مجانا وحصريا

حمل الآن

المراجعة رقم (1)

الترم الاول





لغة البرمجة JavaScript

- **تعريف البرمجة:** هي كتابة أوامر وتعليمات إلى الكمبيوتر ليقوم بتنفيذها.

- توجد لغات برمجة عديدة وتختلف حسب الغرض منها. لذلك نستخدم مزيجاً منها.

JavaScript إنشاء الحركة/التفاعل **HTML + CSS + JavaScript**	CSS إنشاء المظهر **HTML + CSS**	HTML بناء الهيكل **HTML**

- HTML هى لغة ترميز، تُستخدم لبناء **هيكل صفحات الويب** ومحتواها مثل النصوص والصور إلخ.
- CSS هي لغة أنماط تُستخدم لتنسيق صفحات الويب المصممة بلغة HTML.

بمعنى آخر: HTML تبني هيكل صفحة الويب، و CSS تضيف للصفحة الجمال والتنسيق.

لاحظ: HTML و CSS لا تُعدان من لغات البرمجة.

- لغة JavaScript هى لغة تفاعلية.

مثال لصفحة تفاعلية: المكتبة الإلكترونية - وزارة التربية والتعليم studentbooks.moe.gov.eg

- الصفحات التفاعلية تقوم بـ:

عرض الرسائل للمستخدم، إجراء العمليات الحسابية المختلفة، وتشغيل الألعاب.

كتابة كود JavaScript:

فى محرر الأكواد "Code Editor" مثل:

- Visual Studio Code أو Notepad++.
- أو فى صفحة HTML باستخدام الوسم <script> ... <script/>

مثال كتابة كود لطباعة "رسالة ترحيب" فى وحدة التحكم باستخدام **JavaScript:**

```
<html>
    <script>
        console.log("Welcome to the world of programming");
    </script>
</html>
console.log("Welcome to the world of programming");
```

لاحظ: هذا السطر يطبع الرسالة داخل **وحدة التحكم (Console)** في المتصفح.

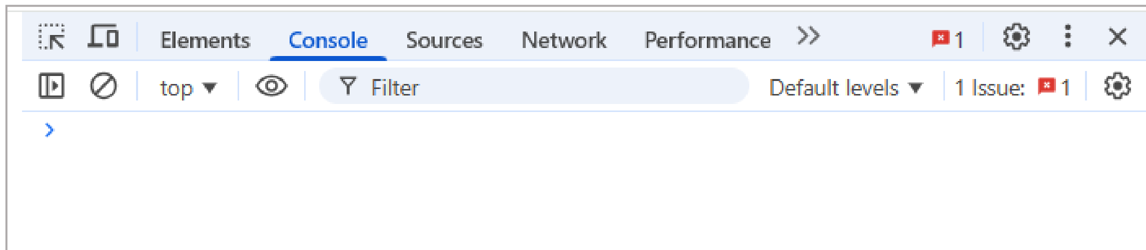
وحدة التحكم console

هي مكان في المتصفح يعرض نتائج الأوامر ويعرض الأخطاء كذلك.

عرض وحدة التحكم:

اضغط على الأزرار: **Ctrl + Shift + I**

ثم اختر وحدة التحكم: **"console"**



مثال كود Javascript: لعرض كلمة "الفائز" فى لوحة التحكم.

```
console.log("الفائز");
```

الملاحظات (التعليقات):

ترك "ملاحظة: فى الكود يسمى "تعليق"

ويُستخدم التعليق (لشرح الكود داخل الكود) لأنه لا ينفذ.

يتم إضافة تعليق إما فى: ← سطر واحد أو في ← عدة أسطر.

❶ إضافة تعليق في سطر واحد:

يسبق التعليق بـ // (شرطتين مائلتين) فى بداية السطر.

مثال:

```
// يسبق التعليق شرطتين مائلتين فى بداية كل سطر
console.log;("مرحبا بالعالم")
```

❷ إضافة تعليق متعدد الأسطر:

نستخدم علامة النجمة و الشرطة المائلة /* .... */ .

مثال:

```
/*
هذا التعليق ممتدد على عدة أسطر
يبدأ التعليق بـ */ وينتهى بـ /* وكل ما بينهما لا يتم تنفيذه
*/
```

التعامل مع النصوص:

يتم دمج النصوص فى JavaScript بعدة طرق، أشهرها استخدام عامل الجمع **+**.
يستخدم عامل الجمع + لربط سلسلتين نصيتين أو أكثر.

مثال 1: دمج النصوص.

```
console.log("مرحبًا بك" + "في جافا سكريبت");
```

مثال 2: دمج النصوص: باستخدام عامل الجمع "+"

```
let name = "JavaScript";
console.log("Hello" + name + "!");
```

console.log : دالة تُستخدم لعرض النتائج في وحدة التحكم (Console)

التعامل مع الأعداد:

باستخدام العمليات الحسابية الأساسية.

العملية	الجمع	الطرح	الضرب	القسمة	باق لبقسمة
الرمز	+	-	*	/	%
مثال	4 = 2 + 2	1 = 1 – 2	4 = 2 * 2	2 = 2 / 4	1 = 2 % 5

مثال: إجراء عمليات حسابية:

```
console.log(2+2);
console.log(2-1);
console.log(2*2);
console.log(4/2);
console.log(5%2);
```

دمج الأعداد والنصوص فى JavaScrip:

```
console.log(2 + 2); // ناتج جمع 2+2
```

أرقام   نص

الإعلان عن المتغيرات

- يمكن الإعلان عن متغير في JavaScript باستخدام الكلمات المفتاحية **let** أو **const**
- **let** تُستخدم للمتغيرات التي يمكن تغيير قيمتها لاحقاً.
- **const** تُستخدم للمتغيرات التي لا يمكن تغيير قيمتها بعد تعيينها لأول مرة.

إعلانات المتغيرات وإسناد القيم:

أولاً: تعريف المتغير: المتغير هو مكان لتخزين البيانات في الذاكرة.

إعلان المتغير age : بستخدام **let**.

javascript

```
let age;    // اعلان عن متغير باسم age دون إسناد قيمة له
```

إسناد قيمة للمتغير age :

```
age = 15;  // تعيين القيمة 15 للمتغير age
```

إعلان المتغير age وإسناد قيمة له.

```
let age = 15;
```

لاحظ أوامر javascrit **تنتهي بالفاصلة المنقوطة** "**;**".

إعلان المتغير name وإسناد قيمة له.

```
let name = "Hesham"; // إنشاء متغير اسمه name وتخزين القيمة "Hesham" بداخله
```

لاحظ : قيمة المتغير name النصية بين علامتى تنصيص

مثال على إعلان المتغيرات:

```
let x = 2;     // إعلان عن المتغير x واسناد القيمة 2 له
let y = 4;
let sum = x * y;     // إعلان عن متغير واسناد ناتج ضرب قيم متغيرات أخرى له
console.log("المجموع=" + sum)
```

لاحظ فى الكود تمت كتابة: "المجموع=" بين علامتى تنصيص لكى تظهر فى لوحة التحكم كما هى.

جملة if

- جملة if في JavaScript تُستخدم لاتخاذ قرارات برمجية.
حيث تنفذ كتلة من التعليمات البرمجية فقط إذا كان الشرط المحدد صحيحًا (true).

تتكون جملة **if** من:

كلمة **if** يتبعها شرط بين قوسين **( )**،

ثم التعليمات البرمجية بين قوسين معقوفين **{ }**.

إذا كان الشرط خاطئاً (false)، يتم تخطي الكود الموجود داخل الأقواس المعقوفة.

مثال:

```
let grade = 10;            // اعلان متغير اسمه grade وإسناد القيمة 10 له
if (grade >= 5 ) {         // الشرط بين القوسين
   console.log("ناجح");    // كود تحقق الشرط
 }
```

جملة if else

- عبارة if...else في JavaScript تستخدم لتنفيذ كتلتين مختلفتين من التعليمات البرمجية بناءً على ما إذا كان الشرط المحدد صحيحًا أم خطأ.

تتكون جملة **if…else** من:

كلمة **if** يتبعها شرط بين قوسين **( )**،

ثم تعليمات برمجية تُ نفذ إذا كان الشرط صحيحاً (true).

تليها كلمة **else** (اختياري)

وتعليمات برمجية تُنفذ إذا كان الشرط خاطئاً (false).

ويمكن إضافة جمل else if للتحقق من شروط متعددة.

مثال 1:

```
let age = 25;   //  إعلان متغير اسمه age وتخزين القيمة 25 فيه
if (age >= 18) {
  console.log("أنت مؤهل للتصويت");  // يتم تنفيذ هذا السطر إذا كان الشرط صحيح
} else {
  console.log("أنت غير مؤهل للتصويت");  // يتم تنفيذ هذا السطر إذا كان الشرط خطأ
}
```

مثال 2:

```
let temp = 30;
if (temp > 35) {
console.log("الجو حار");
} else {
console.log("الجو معتدل");
}
```

شرح الكود فى المثال 2:

let temp = 30;

- إنشاء متغير اسمه temp وتخزين القيمة 30 فيه.

if (temp > 35)

- الشرط يتحقق: هل درجة الحرارة أكبر من 35؟

إذا كان صحيحًا ← ينفذ الكود داخل {} الأول ويطبع "الجو حار".

إذا كان خطأً ← ينتقل إلى else ويطبع "الجو معتدل".

- **النتيجة عند التنفيذ في الـ Console:** الجو معتدل

جملة if else if

- عبارة if...else في JavaScript تستخدم لتنفيذ العديد من التعليمات البرمجية المختلفة بناءً على ما إذا كان الشرط المحدد صحيحًا أم خطأ.

مثال :

```
let grade = 85;
if (grade >= 90) {
  console.log("excellent");
} else if (grade >= 75) {
  console.log("very good");
} else {
console.log("Need more effort");
}
```

شرح الكود فى المثال:

let grade = 85;

- إنشاء متغير اسمه grade وتخزين القيمة 85 فيه.

if (grade >= 90)

- يتحقق: هل الدرجة 90 أو أكثر؟

← لا ، لأن 85 أصغر من 90 .

else if (grade >= 75)

- يتحقق: هل الدرجة 75 أو أكثر؟

← نعم ، لأن 85 أكبر من 75، إذن ينفذ السطر very good

- **النتيجة عند التنفيذ في الـ Console:** very good

المعاملات المنطقية

- المعاملات المنطقية في JavaScript هي:

&& (و AND)، || (أو OR)، ! (ليس NOT)

تُستخدم المعاملات المنطقية لدمج قيم منطقية وتحديد العلاقة بين المتغيرات، حيث:

يُرجع العامل المنطقى && القيمة "صحيح" فقط إذا كانت جميع العبارات صحيحة،

ويُرجع العامل المنطقى || "خطأ" إذا كانت إحدى العبارات صحيحة على الأقل،

أما العامل المنطقى ! يعكس القيمة المنطقية.

مثال 1:

```
(x < 10 && y > 12) // العامل && (AND) يجعل الناتج صحيحاً فقط إذا تحقق الشرطان معاً
(x < 10 || y > 12)  // العامل || (OR) يجعل الناتج صحيحاً إذا تحقق أحد الشرطين أو كلاهما
! (x < 10)            //  العامل ! (NOT) يعكس نتيجة الشرط، إذا كان صحيحاً يصبح أخطأ والعكس
```

مثال 2:

```
let x = 8;      // الشرط صحيح لأن قيمة المتغير أكبر من 5
let y = 15;    // الشرط صحيح لأن قيمة المتغير أقل من 20
 if (x > 5 && y < 20) {
   console.log("الشرطين صحيحين");
} else {
console.log("أحد الشرطين أو كلاهما غير صحيح");
}
```

كسر السطر في JavaScript:

- يمكن كسر الأسطر باستخدام رمز السطر الجديد \n داخل السلاسل النصية.
- استخدام \n لإدراج فاصل سطر داخل سلسلة نصية.

أمثلة:

```
console.log("البرمجة\nعالم");
console.log("الأول الثانوى\nالصف");
const message = "Hello\nWorld";
console.log(message);
```

النتيجة عند التنفيذ في الـ Console:

عالم
البرمجة
الصف
الأول الثانوى
Hello
World

العمليات التكرارية

جملة for

- تُستخدم for فى الحلقات التكرارية في JavaScript لتنفيذ تعليمات برمجية عدة مرات.

العمليات التكرارية for وزيادة "increment" عداد الحلقة التكرارية:

مثال: حلقة for عند العلم مسبقًا بعدد المرات التي تريد تكرارها.
طباعة الأعداد من 0 إلى 4.

```
for (let i = 0;   i < 5;  i++) {
     console.log(i);
 }
```

شرح أجزاء الكود:

		النتيجة عند التنفيذ:
let i = 0	يقوم بتهيئة متغير العداد i بالقيمة 0	0
i < 5;	يستمر الشرط طالما أن قيمة i أقل من 5	1
i++	تعني **زيادة** قيمة متغير العداد i بمقدار 1 بعد كل مرة تكرار.	2
console.log(i);	يطبع قيمة i فى كل مرة	3 4

الدالة Function

- الدالة (Function): هى مجموعة من الأوامر تنفذ عند استدعائها.
- يمكن استخدام دوال JavaScript لإنشاء تعليمات برمجية قابلة لإعادة الاستخدام لتنفيذ مهمة معينة.

يتم تعريف الدالة باستخدام كلمة **function** متبوعة باسم الدالة،
ثم أقواس ( ) تحتوي على المعاملات (إن وجدت)،
ثم أقواس معقوفة { } تحتوي على كود الدالة.
يمكنك استدعاء الدالة بتكرار اسمها متبوعاً بالأقواس ( ).

مثال 1: إعلان دالة باسم greet

```
function greet() {
 console.log("تحية لك");
 }
greet ();  // لاستدعاء الدالة نكتب اسمها متبوعاً بالأقواس
```

النتيجة عند التنفيذ:

تحية لك

مثال 2: إعلان دالة باسم addNumbers تقوم بجمع رقمين

```
javascript
function addNumbers(a, b) {
return a + b;
}
let result = addNumbers(5, 3);
console.log(result);   // 8 سيقوم بطباعة
```

النتيجة في الـ Console:

8

الدالة Math.floor()

قطع الكسر (إزالة الجزء العشري من العدد).

- الدالة Math.floor() هي دالة تُرجع أكبر عدد صحيح أصغر من أو يساوي الرقم المُعطى (أي تُقرّب الرقم للأسفل)، مما يؤدي إلى "قطع" الجزء العشرى.

على سبيل المثال:

القيمة الراجعة من الدالة	استخدام الدالة مع الأرقام الموجبة
الناتج: 5 //	console.log(Math.floor(5.6));
الناتج: 5 //	console.log(Math.floor(5.9));
الناتج 7 لأنه عدد صحيح بالفعل //	console.log(Math.floor(7));

القيمة الراجعة من الدالة	استخدام الدالة مع الأرقام السالبة
الناتج: -3 //	console.log(Math.floor(-2.5));
الناتج: -11 //	console.log(Math.floor(-10.5));

استخدام الدالة:

الصيغة: Math.floor(x) حيث (x) هو الرقم المطلوب تقريبه.

مثال

```
console.log(Math.floor(5.9));  // 5 القيمة الراجعة
```

شرح الكود: Math.floor(x) ← تقوم بتقريب الرقم x إلى أقرب عدد صحيح أقل منه أو يساويه.

الناتج في وحدة التحكم (Console): 5

الدالة Math.random()

العمل مع الأرقام العشوائية:

- الدالة Math.random() : تستخدم لإنشاء أرقام صحيحة عشوائية ضمن نطاق محدد، مثل اختيار عنصر عشوائي من مصفوفة.

مثال 1

إنشاء دالة عشوائية: وظيفتها إعطاء رقم عشوائى بين 0 ، 1 .

javascript

```
console.log(Math.random()); // كل مرة تنفذ الكود، ستحصل على رقم مختلف تقريبًا
```

مثال 2

إنشاء دالة عشوائية: وظيفتها إعطاء رقم عشوائى بين 1 ، 10 .

javascript

```
console.log(Math.floor(Math.random() * 10) + 1);
```

أنواع البيانات

- بعض أنواع البيانات في JavaScript.

النوع		مثال
رقمى	Number	2 , 3.14
نصى	String	“AL-Faez”
منطقى	Boolean	True , False
كائن	Object	{name: “Laptop”}
مصفوفة	Array	[1, 2, 3]

تحديد نوع البيانات فى JavaScript

- تحديد نوع البيانات في متغير أو تعبير معين باستخدام **typeof** .

مثال:

```
let x = 7;
console.log(typeof x);   // number
```

مقارنة القيم والأنواع

- تُقارن JavaScript القيم إما باستخدام == التي تقوم بتحويل الأنواع تلقائيًا، أو باستخدام === التي تتطلب أن يكون كل من القيمة والنوع متطابقين.

- عند مقارنة أنواع مختلفة باستخدام ==، تُحوّل JavaScript القيم إلى نوع واحد لإجراء المقارنة.
- وعند المقارنة باستخدام === فتتجنب هذا التحويل وتُرجع false إذا كانت القيمتان من نوعين مختلفين، لذلك هى الطريقة الأكثر أماناً.

الخلاصة:

المعامل	الاستخدام	أمثلة	الناتج
==	مقارنة القيم فقط	“2” == 2	صحيح
===	مقارنة القيم والأنواع	“2” === 2	خطأ

التحويل بين الأنواع

- الدالة ()Number: تحول سلسلة نصية إلى رقم (صحيح أو عشري).
- تُرجع (Not a Number) NaN إذا لم يكن التحويل ممكنًا، أو 0 للسلسلة النصية الفارغة "".

التحويل من نص إلى رقم باستخدام الدالة ()Number :

```
javascript
Number ("5"); // 5
```

شرح الكود:

"5" ← نص (string) يمثل الرقم 5.

Number ("5") ← يحوّل النص "5" إلى رقم فعلي من نوع number.

بعد التحويل يمكن استخدام الرقم 5 في العمليات الحسابية:

- تُستخدم الدالة ()String لتحويل أنواع بيانات أخرى إلى سلسلة نصية.

التحويل من رقم إلى نص باستخدام الدالة ()String :

```
javascript
String(7);  // "7"
```

المصفوفة (Array)

- المصفوفة (Array) في جافا سكريبت: تسمح بتخزين قيم متعددة في متغير واحد، حيث يُمكن الوصول إلى كل قيمة باستخدام فهرسها (index) الرقمي الذي يبدأ من 0.
تُعرّف المصفوفة باستخدام الأقواس المربعة [] أو الدالة ()new Array.

مثال:

```
let countries = ["الإمارات" ,"السعودية" ,"مصر "];
console.log(countries[0]); // مصر
console.log(countries[1]); // السعودية
console.log(countries[2]); // الإمارات
```

شرح كود المصفوفة:

```
let countries= ["الإمارات" ,"السعودية" ,"مصر "];
```

السطر الأول: إنشاء مصفوفة اسمها countries

تحتوي على 3 عناصر نصية (String).

1- مصر

2- السعودية

3- الإمارات

كل عنصر له رقم تسلسلى (فهرس) يبدأ من الصفر.

```
console.log(countries[0]); // مصر
```

السطر الثانى: يطبع العنصر رقم 0 من الصفوفة وهو "مصر".

```
console.log(countries[1]); // السعودية
```

السطر الثالث: يطبع العنصر رقم 1 من الصفوفة وهو " السعودية ".

```
console.log(countries[2]); // الإمارات
```

السطر الرابع: يطبع العنصر رقم 2 من الصفوفة وهو "الإمارات".

الناتج النهائي:

مصر

السعودية

الإمارات

مثال: استخدام جملة التكرار for مع عنا صر المصفوفة.

```
javascript
let numbers = [5, 15, 25, 35, 45];
for (let i = 0; i < numbers.length;  i++) {
console.log("رقم العنصر" + i + "هو" + numbers[i]);
}
```

شرح كود المصفوفة:

let numbers = [5, 15, 25, 35, 45];

السطر الأول: إنشاء مصفوفة اسمها numbers تحتوي على 5 عناصر: 5، 15، 25 ، 35 ، 45.

for (let i = 0; i < numbers.length; i++) {

السطر الثانى: الحلقة التكرارية for :

let i = 0 : يبدأ العد من الصفر (لأن أول عنصر في المصفوفة رقمه 0).

i < numbers.length :

تستمر حلقة التكرار طالما أن **i** أصغر من عدد عناصر المصفوفة (numbers.length = 3)

i++ : بعد كل دورة تكرار ، تزداد قيمة المتغير **i** بمقدار **1**.

console.log("رقم العنصر" + i + "هو" + numbers[i]);

السطر الثالث: في كل دورة، يطبع البرنامج جملة تحتوي على:

رقم العنصر(i) والقيمة الموجودة في هذا العنصر(numbers[i])

الدالة ()console.log تستخدم لطباعة النتائج في نافذة Console داخل المتصفح.

الناتج النهائى:

رقم العنصر 0 هو 5

رقم العنصر 1هو 15

رقم العنصر 2 هو 25

رقم العنصر 1هو 35

رقم العنصر 2 هو 45

نماذج أسئلة تدريبية

السؤال الأول: اختر الإجابة الصحيحة:

1- الكود التالي يطبع قيمة المتغير x بعد تنفيذ الأمر x = x + 2
إذا كانت قيمة x تساوي 5، ما ستكون القيمة الجديدة؟

أ) 5 ب) 6 ج) 7 د) 8

2- ...... هى عبارة عن مجموعة أوامر تنفذ عند استدعائها.

أ) الدالة (function) ب) && ج) // د) ++

3- أى من العبارات التالية تُستخدم لتكرار عدد محدد من المرات؟

أ) for ب) if ج) switch د) try

4- ناتج الكود التالى: Math.floor(5.6) ، هو......

أ) 6 ب) 5 ج) 56 د) 65

5- نستخدم الرمز // فى ...... السطر الواحد في كود JavaScript.

أ) حذف ب) نسخ ج) حفظ د) كتابة تعليقات

6- نستخدم بيانات من النوع ...... لتخزين سلسلة نصية.

أ) Boolean ب) String
ج) Number د) Array

7- لعرض رسالة على شاشة المتصفح نستخدم الأمر......

أ) console ب) console.log
ج) show د) display

8- الغرض من البرمجة هو ......

أ) كتابة نصوص فقط ب) عرض الأرقام
ج) تصميم الصور د) كتابة أوامر ليقوم الكمبيوتر بتنفيذها

9- ...... هي الكلمة المحجوزة التي تُستخدم لإعلان متغير يمكن تغيير قيمته.

أ) let ب) variable
ج) var د) const

10- الهدف من أمر return في الدالة (Function) هو ......

أ) إيقاف البرنامج ب) تكرار حلقة
ج) إرجاع قيمة من الدالة د) طباعة متغير

11- الرمز (+) في JavaScript يُستخدم في......

أ) الجمع فقط ب) الطرح فقط

ج) الجمع ودمج النصوص د) القسمة فقط

12- ...... هو رمز التعليق في سطر واحد.

أ) /* ب) &

ج) % د) /* ... */

13- ...... هو الرمز المستخدم لكتابة تعليق في أكثر من سطر .

أ) /* ب) &

ج) % د) /* ... */

14- فى الكود التالى: console.log("Hello\nWorld");

الرمز \n يُستخدم في ......

أ) كسر السطر ب) إدراج سطر جديد داخل النص

ج) الانتقال إلى سطر جديد د) كل ما سبق

15- جملة ...... تستخدم في تكرار تنفيذ الأوامر عددًا معينًا من المرات.

أ) for ب) while

ج) if د) else if

16- الرمز ++ يُستخدم في ...... بمقدار 1.

أ) نقصان المتغير ب) زيادة المتغير

ج) التكرار د) رجوع القيمة

17- نستخدم المتغير في ......

أ) النسخ ب) الحفظ

ج) الرسم د) تخزين البيانات

18- ...... هو إجراء يُعزز أمان حسابك.

أ) استخدام كلمة مرور بسيطة ب) مشاركة كلمة المرور

ج) كتابة كلمة المرور على ورقة عامة د) تفعيل المصادقة الثنائية

19- ناتج الكود التالى هو.... console.log(4 + 2);

أ) 35 ب) 53

ج) 6 د) خطأ

20- ...... هو رمز الضرب في العمليات الحسابية.

أ) / ب) *

ج) x د) @

21- عملية دمج النصوص يُطلق عليها ......

أ) الجمع   ب) الحساب

ج) التجميع   د) المجموع

22- نوع العملية التي تقوم بمقارنة قيمتين هى عملية ......

أ) حسابية   ب) إدخال

ج) إخراج   د) منطقية

23- إذا أردنا تخزين مجموعة من الأسماء في متغير واحد قابلة للترتيب والبحث، أي البنى نستخدم؟

أ) Array   ب) String

ج) Number   د) Boolean

24- أي من التالي يعتبر مثالاً لتطبيق الذكاء الاصطناعي؟

أ) مفكرة نصية   ب) التعرف على البصمة أو الوجه

ج) متصفح ويب   د) آلة حاسبة بسيطة

25- ما نتيجة العبارة المنطقية: (5 > 3) && (2 == 2)؟

أ) false   ب) undefined

ج) خطأ نحوى   د) true

26- نستخدم الرمز && في ......

أ) العمليات المنطقية (AND)   ب) العمليات المنطقية (OR)

ج) repeat   د) return

27- ناتج الكود التالى: if (x > 10 && y < 10)

إذا كانت x=12 ، y=8

أ) false   ب) true

ج) خطأ   د) لاشئ مما سبق

28- نستخدم الدوال في البرمجة بهدف ......

أ) تنظيم الكود   ب) إعادة استخدام الكود

ج) كل من أ ، ب   د) تغيير نوع البيانات

29- الناتج من الكود التالى: console.log("5" + 2); ، **هو......**

أ) 7   ب) 52

ج) خطأ   د) 3

30- لكتابة كود JavaScript نستخدم ......

أ) برنامج التشغيل windows   ب) محرر الكود

ج) برنامج Word   د) برنامج Excel

31- الأمر typeof true يعطي الناتج .....

أ) text ب) string

ج) boolean د) number

32- المعامل === ......

أ) يقارن القيمة فقط ب) يقارن النوع فقط

ج) يقارن القيمة والنوع معاً د) لا شئ مما سبق

33- المعامل == ......

أ) لا يقارن شئ ب) يقارن النوع فقط

ج) يقارن القيمة بعد تحويل النوع د) لا شئ مما سبق

34- الكود ()Math.random ينتج رقمًا عشوائيًا بين ......

أ) 1,0 ب) 10,10

ج) 100,10 د) 1,10

35- النوع "......" يمثل قيمتين منطقيتين هما true أو false .

أ) string ب) text

ج) boolean د) number

36- أي من الأكواد التالية يُنشئ مصفوفة صحيحة؟

أ) let number = {10, 20, 30} ب) let number = [10, 20, 30]

ج) let number = (10, 20, 30) د) let number = <<10, 20, 30>>

37- الأقواس …… هي التي تُستخدم لإنشاء المصفوفات (Arrays) في JavaScript.

أ) المربعة [ ] ب) الدائرية ( )

ج) المعقوفة { } د) الزاويّة <>

38- الأمر typeof 6 يعطي الناتج .....

أ) text ب) string

ج) boolean د) number

39- الأمر "typeof "Hello يعطي الناتج .....

أ) text ب) string

ج) boolean د) number

40- وحدة التحكم Console تُستخدم لعرض ......

أ) تقديمي ب) ناتج الكود

ج) الأخطاء د) كل من ب ، ج

41 لإعلان متغير باسم name نكتب الكود التالى ......

أ) let name
ب) let = name
ج) consol name
د) name = let

42- العامل المنطقي (&&) يُرجع true فقط إذا كانت العبارتان ...... معًا.

أ) كاملتين
ب) جملة اسمية
ج) صحيحتين
د) غير صحيحتين

43- ...... هى الجملة الشرطية التي تنفذ أمرًا عند تحقق شرط فقط.

أ) for
ب) if
ج) else
د) else if

44- نستخدم ...... عند التعامل مع أكثر من حالة شرطية.

أ) for
ب) if
ج) else
د) else if

45- أسلوب التعلم الذي يعتمد على تزويد النموذج بكميات كبيرة من البيانات ويسمّى ......

أ) التعلم الآلي
ب) Machine Learning
ج) كل من أ ، ب
د) كل ما سبق

السؤال الثانى: ضع علامة (✓) أو علامة (X) أمام كل ما يلى:

1- لغة JavaScript لا تُستخدم في برمجة الويب. ( )
2- الحلقات التكرارية تُستخدم لتنفيذ مجموعة من الأوامر مرة واحدة فقط. ( )
3- الذكاء الاصطناعي لا يحتاج بيانات للتعلم. ( )
4- كلمة المرور القوية هي التي تحتوي على حروف فقط. ( )
5- المتغير يمكن تغيير قيمته خلال تنفيذ البرنامج. ( )
6- الدالة لا يمكنها أن تستقبل أي معاملات. ( )
7- المصفوفة (Array) يمكنها تخزين عناصر من أنواع مختلفة فى لغة JavaScript. ( )
8- الهندسة الاجتماعية هي أحد أساليب الهجوم السيبراني. ( )
9- حقوق الملكية الفكرية لا تنطبق على البرمجيات. ( )
10- واجهة المستخدم (UI) عنصر مهم فقط في البرامج الكبيرة، وليس للمشروعات الصغيرة. ( )
11- لطلب إدخال من المستخدم ، نستخدم الدالة Ask . ( )
12- لطلب إدخال من المستخدم ، نستخدم الدالة prompt(). ( )

إجابة نماذج الأسئلة تدريبية

السؤال الأول:

(1) ج (2) أ (3) أ (4) ب (5) د (6) ب (7) ب (8) د (9) أ (10) ج

(11) ج (12) أ (13) د (14) د (15) أ (16) ب (17) د (18) د (19) ج (20) ب

(21) ج (22) د (23) أ (24) ب (25) د (26) أ (27) ب (28) ج (29) ب (30) ب

(31) ج (32) ج (33) ج (34) أ (35) ج (36) ب (37) أ (38) د (39) ب (40) د

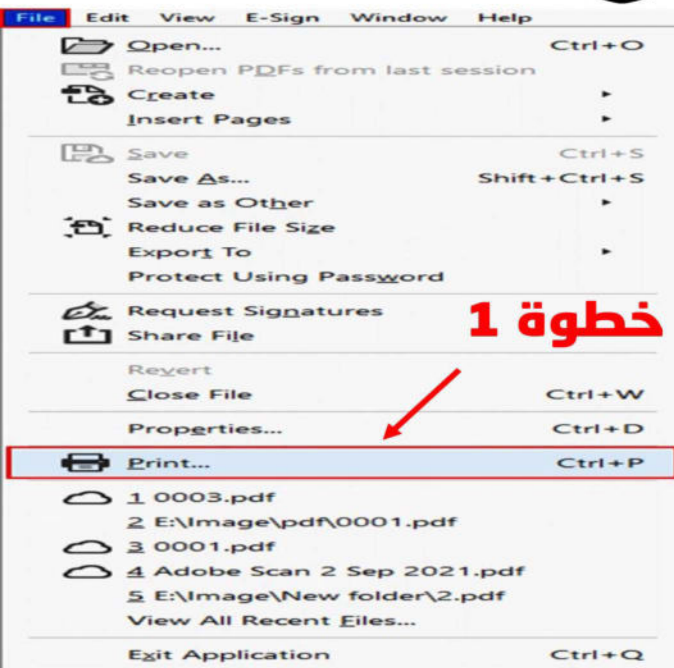
(41) أ (42) ج (43) ب (44) د (45) د

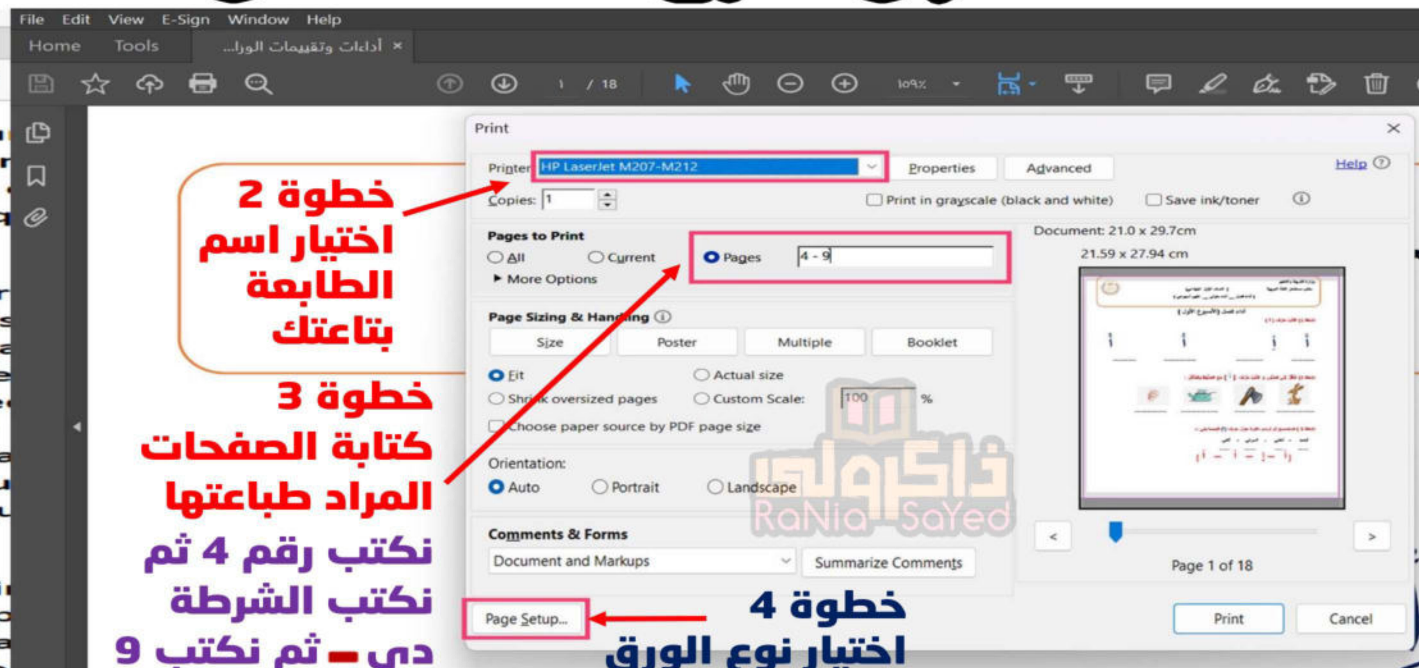
السؤال الثانى:

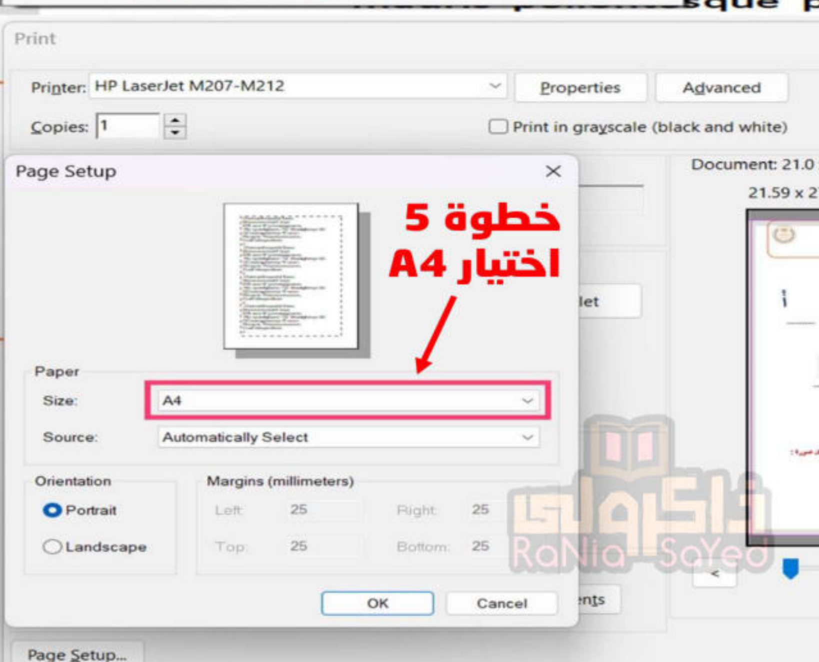
(1) خطأ: JavaScript تُستخدم على نطاق واسع في الويب.
(2) خطأ: الحلقات تُستخدم للتكرار أكثر من مرة.
(3) خطأ: معظم أساليب التعلم الآلي تحتاج بيانات للتدريب.
(4) خطأ كلمة مرور قوية تجمع حروفًا كبيرة وصغيرة وأرقامًا ورموزًا.
(5) صح
(6) خطأ: الدوال يمكنها استقبال معاملات (Parameters)
(7) صح
(8) صح
(9) خطأ: للبرمجيات حقوق ملكية فكرية وقوانين تحميها.
(10) خطأ: واجهة المستخدم مهمة دائمًا لتفاعل المستخدمين.

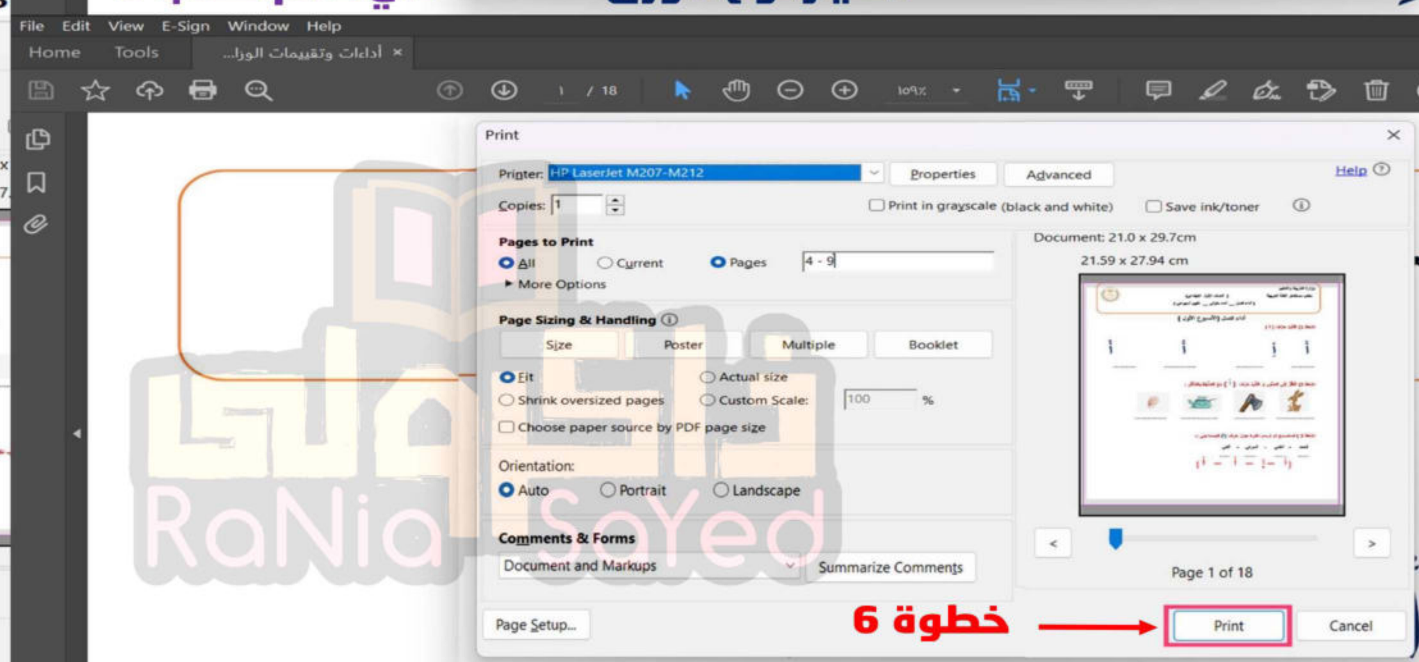
كيفية طباعة صفحات معينة من ملف معين

مثلا ازاى نطبع الصفحات من صفحة 4 الى صفحة 9









حمل الآن

مجانا وحصريا

المراجعة رقم (2)

الترم الاول





Chapter 1 : Introduction

الفصل الأول : مقدمة

What is Programming [1]

ما هي البرمجة [١]

1 what can you do with programming ?

١ ما الذي يمكنك فعله بالبرمجة ؟

Programming allows you to create games, websites, apps, etc.

تتيح لك البرمجة إنشاء ألعاب ومواقع ويب وتطبيقات وما إلى ذلك

What is Programming [2]

ما هي البرمجة [٢]

1 what is programming ?

١ ما هي البرمجة ؟

Programming is the process of writing instructions or code for a computer to successfully perform specific tasks, or in short, it is the ability to create programs

البرمجة هي عملية كتابة التعليمات أو الأكواد للحاسوب بهدف تنفيذ مهمات معينة بشكل ناجح أو باختصار شديد هي القدرة على إنشاء برامج

2 what is the program ?

٢ ما هو البرنامج ؟

- A program is a structured set or sequence of steps or instructions that are executed in order to accomplish a specific task.
- A program could be a calculator that adds two numbers, a washing machine program, or a daily program for a trip.
- written in programming language, not in ordinary letters

- البرنامج هو مجموعة أو سلسلة منظمة من الخطوات أو الأوامر بيتم تنفيذها بالترتيب لإنجاز مهمة محددة
- البرنامج ممكن يكون آلة حاسبة تجمع رقمين أو برنامج تشغيل غسالة أو برنامج يومي لرحلة في مكان ما
- يكتب بلغات البرمجة، وليس بالأحرف العادية

2 what is Programming language ?

٢ ما هي لغة البرمجة ؟

Special language that computers can understand

لغة خاصة يمكن لأجهزة الكمبيوتر فهمها

3 how many programming languages ?

كم عدد لغات البرمجة ؟ ٣

there are more than 100 programming languages, and here we will learn three types : HTML - CSS - JavaScript

هناك أكثر من ١٠٠ لغة برمجة، وهنا سوف نتعلم ثلاثة أنواع: HTML - CSS - JavaScript

HTML

- Hyper Text Markup Language
- Used to build the framework
- markup language (not a programming language)

- لغة ترميز النص التشعبي
- تُستخدم لبناء إطار العمل
- لغة ترميز (ليست لغة برمجة)

CSS

- Cascading Style Sheets
- used to create appearance
- stylesheet language (not a programming language)

- أوراق الأنماط المتتالية
- تُستخدم لإنشاء مظهر
- لغة أوراق الأنماط (ليست لغة برمجة)

JavaScript

- It adds logic, interactivity, and behavior
- programming language of the web

- تضيف المنطق والتفاعلية والسلوك
- لغة برمجة الويب

How to use an editor

كيفية استخدام المحرر

1 about text editors

حول محرري النصوص ١

A text editor is a black screen on which programs are written, it is also called an editor for short and can also be used to write programs

محرر النصوص هو عبارة عن شاشة سوداء يتم كتابة البرامج عليها، ويسمى أيضًا محررًا للاختصار ويمكن استخدامه أيضًا لكتابة البرامج

2 console.log(" "); ٢

It is a function that lets you print messages or values to the console

إنها وظيفة تتيح لك طباعة الرسائل أو القيم في وحدة التحكم

3 console.log(" "); ٣

When you write a program in the code editor and run it, the computer shows the result in a place called the console

عندما تكتب برنامجًا في محرر الكود وتقوم بتشغيله، يعرض الكمبيوتر النتيجة في مكان يسمى وحدة التحكم

4 console.log(" "); ٤

- The dot (.) means we are using a function that belongs to the console
- The word "log" means "write" or "show"
- characters enclosed in double quotation marks (" ") will be output to the console
- The semicolon (;) is used to mark the end of a statement

- النقطة (.) تعني أننا نستخدم دالة تابعة لوحدة التحكم
- كلمة "log" تعني "write" أو "show"
- سيتم طباعة الأحرف الموجودة بين علامتي اقتباس (" ") إلى وحدة التحكم
- تُستخدم الفاصلة المنقوطة (;) لتمييز نهاية العبارة

5 how to type into a text editor ٥ كيفية الكتابة في محرر النصوص

Inside the editor, you can write something like (Hello, Good morning) :

داخل المحرر، يمكنك كتابة شيء مثل (Hello, Good morning) :

main.js
```
console.log("Hello");
console.log("Good morning");
```

Output
```
Hello
Good morning
```

This line tells the computer to show "Hello" and "Good morning" in the console

يخبر هذا السطر الكمبيوتر بطباعة "Hello" و "Good morning" في وحدة التحكم

Check your typing carefully

- JavaScript is case-sensitive, so "console" and "Console" are not the same
- Make sure you add quotes (" ") around text
- Don't forget the semicolon (;) at the end

تحقق من كتابتك بعناية

- JavaScript حساسة لحالة الأحرف، لذا فإن "console" و"Console" مختلفان
- تأكد من إضافة علامتي اقتباس (" ") حول النص
- لا تنسَ الفاصلة المنقوطة (;) في النهاية

Console

وحدة التحكم

1 What is console ?

Its the black screen on the right that helps you see what your code is doing

١ ما هي وحدة التحكم ؟

إنها الشاشة السوداء على اليمين التي تساعدك على رؤية ما يفعله الكود الخاص بك

2 Output

it is a programming term means display something on the console

٢ الناتج

هو مصطلح برمجي يعني عرض شيء ما على وحدة التحكم

3 Errors

if there is any mistake in the program, an error will be displayed

main.js

```
console.log(see you tomorrow)
```

* Double quotation marks are missing in the program

* Simicolon can be ommited like the Period (.), but if you add or omit the (.) in a sentence it will be difficult to read

٣ الأخطاء

في حالة وجود أي خطأ في البرنامج سيتم عرض الخطأ

Output

```
Uncaught SyntaxError: ............
```

* علامات الاقتباس مفقودة في البرنامج

* يمكن حذف الفاصلة المنقوطة مثل النقطة (.)، ولكن إذا أضفت أو حذفت (.) في جملة، فسيكون من الصعب قراءتها

Comment

التعليق

1 comment

Making a comment is by adding two slashes (//) at the beginning of each line

main.js

```
// greetings for the day
console.log("Good morning");
console.log("Hello");
console.log("Good evening");
```

* it's helpful to leave a comment when you've written a long, complex program

١ التعليق

تتم عملية التعليق عن طريق إضافة شرطتين مائلتين (//) في بداية كل سطر

Output

```
Good morning
Hello
Good evening
```

* من المفيد ترك التعليقات عند كتابة برنامج طويل ومعقد

تعطيل

Comment out

١ تعطيل

1 comment out

إذا وضعت (//) في بداية البرنامج، يمكنك منع تنفيذ هذا البرنامج مؤقتًا

If you put (//) at the beginning of a program, you can temporarily prevent that program from being executed

main.js

```
console.log("Eggs");
// console.log("Apple");
console.log("Milk");
```

Output

```
Eggs
Milk
```

الفصل الثاني : العمليات الحسابية والسلاسل النصية

Chapter 2 : Calculations and strings

كيفية الإخراج إلى وحدة التحكم

How to output to the console

١ ما هي الأشياء التي تجيدها أجهزة الكمبيوتر ؟

1 What are computers good at

أجهزة الكمبيوتر جيدة في الحساب، ويمكنها إنتاج نتائج حسابية حتى لو كانت الأرقام كثيرة أو معقدة

computers are good at calculating, and can produce calculation results even if there are many digits or they are complex

* "الحواسيب الفائقة" هي أجهزة كمبيوتر عالية الأداء يمكنها معالجة كم هائل من العمليات الحسابية في ثانية واحدة، وهي مهمة قد تستغرق من الإنسان عدة أيام لإنجازها

* "Supercomputers" are high-performance computers that can process huge amount of calculations that would take a person several days to do in just one second

main.js
```
console.log("1+1");
```

Output
```
1+1
```

* لا داعي لوضع علامتي اقتباس المزدوجة (" ") لأن الحاسوب يقرأ النص كنص عادي وليس كعملية حسابية

* you dont need to put double quotations (" ") because the computer read it as a normal text not a mathematical calculations

٢ كيفية عرض نتائج الحساب على وحدة التحكم ؟

2 How to output calculation results to the console

لإخراج نتائج الحساب إلى وحدة التحكم، اكتب صيغة الحساب بين قوسين في console.log();

To output calculation results to the console, write the calculation formula in brackets in console.log();

main.js
```
console.log(1+1);
```

Output
```
2
```

طرق الحساب

Calculations Methods

١ العمليات الحسابية الأساسية الأربع

تُسمى عمليات الجمع والطرح والضرب والقسمة مجتمعة بـ "العمليات الحسابية الأساسية الأربع"

1 Four basic arithmetic operations

addition, subtraction, multiplication and division are collectively called "the Four basic arithmetic operations"

٢ المشغلات الحسابية

" + "," - "," * " و " / " تُسمى مجتمعةً "المشغلات الحسابية"

2 arithmetic operators

" + "," - "," * " and " / " are collectively called "arithmetic operators"

main.js

```
// Addition
console.log(432+345);

// Subtraction
console.log(100-50);

// Multiplication
// [5x9] is written as [5*9]
console.log(5*9);

// Division
// [10÷2] is written as [10/2]
console.log(10/2);
```

Output

```
777
50
45
5
```

* في الضرب والقسمة نستخدم رموزًا مختلفة قليلاً، نستخدم النجمة " * " للضرب والشرطة المائلة " / " للقسمة

* In multiplication and division we use slightly different symbols, we use asterisk " * " for multiplication and slash " / " for division

الأقواس: ( )
علامات التنصيص المزدوجة: ""
الفاصلة المنقوطة: ;
النقطة: .
النجمة: *
الشرطة المائلة: /
الشرطة المائلة المزدوجة: //

Parentheses : ( )
Double quotation marks : ""
semicolon : ;
Period / Dot : .
Asterisk : *
Slash : /
Double Slash : //

الدمج

Consolidation

١ الدمج

العلاقة بين الحروف والصيغ الحسابية أو التعبير عنهم في سطر واحد

1 Consolidation

Connection between characters and calculation formulas or expressing them in a single line

ربط الحروف بالصيغ الحسابية

- أحط الأحرف بعلامة "" (علامة اقتباس مزدوجة)
- أحط الصيغ بعلامة ( ) (أقواس)
- استخدم علامة الجمع (+) لربط الأحرف والصيغ

Connecting characters with calculation formulas

- Enclose characters in "" (Double quotation mark)
- Enclose formulas in ( ) (Parantheses)
- use + (Plus) to connect characters and formulas

main.js

```
console.log("Set of 2 coffees " + (250 * 2) + " codycoins");
```

Output

```
Set of 2 coffees 500 codycoins
```

* لطباعة الأحرف والصيغ في سطر واحد، اكتب :
console.log("text" + (formula) + "text");

* To print characters and formulas in one line write :
console.log("text" + (formula) + "text");

ربط الحروف ببعضها البعض

Connecting characters with each other

main.js

```
console.log("Today's recommendation is " + "soup.");
```

Output

```
Today's recommendation is soup.
```

Connecting characters with numbers

ربط الحروف بالارقام

main.js

```
console.log("Limited to " + 30 + " servings.");
```

Output

```
Limited to 30 servings.
```

Practical application

تطبيق عملي

main.js

```
console.log("100 + 200" + " is " + (100 + 200));
```

Output

```
100 + 200 is 300
```

- Enclose characters in "" (Double quotation mark)
- Enclose formulas in ( ) (Parantheses)
- use + (Plus) to connect characters and formulas

- أحط الأحرف بعلامة "" (علامة اقتباس مزدوجة)
- أحط الصيغ بعلامة ( ) (أقواس)
- استخدم علامة الجمع (+) لربط الأحرف والصيغ

Chapter 3 : Variables

الفصل الثالث : المتغيرات

Declarations and Assignments

الإعلانات والتعيينات

1 what is variable ?

A variable is like a box for storing a value and the value inside can be changed

١ ما هي المتغيرات ؟

المتغير يشبه صندوقًا لتخزين قيمة، ويمكن تغيير القيمة الموجودة بداخله

2 How to write variables

To write a variable, you must follow these three steps :

٢ كيفية كتابة المتغيرات

لكتابة متغير، يجب عليك اتباع هذه الخطوات الثلاث :

Variable declaration

variable declaration simply means telling the program that you want to create a box to store a value, it's form is :

let + variable name + ;

Example :

إعلان المتغير

إعلان المتغير يعني ببساطة إعلام البرنامج بأنك تريد إنشاء صندوق لتخزين قيمة، وصيغته هي :

let + اسم المتغير + ;

مثال :

main.js

```
let apple;
```

Output

```
```

Creating a new variable is called declaring in the programming world

- let : This keyword is used to declare a variable
- apple : This is the name of your variable, you can choose any name

إنشاء متغير جديد يُسمى بالإعلان في عالم البرمجة

- let : تُستخدم هذه الكلمة المفتاحية للإعلان عن متغير
- apple : هذا هو اسم متغيرك، يمكنك اختيار أي اسم

تعيين القيمة

تعيين القيمة يعني وضع قيمة داخل متغير، مثل وضع شيء داخل صندوق قمت بإنشائه بالفعل، وصيغته هي:
اسم المتغير = القيمة;
مثال:

Value assignment

Value assignment means putting a value inside a variable, like placing something inside a box you already created, it's form is : variable name = value;
Example :

```
main.js
let apple;
apple = 0;
```

```
Output
```

وضع قيمة مثل 0 في متغير بهذه الطريقة يُسمى التعيين.

- "0" هي القيمة المخزنة في المتغير.
- "=" هذا هو عامل التعيين. يعني "ضع القيمة في المتغير."

putting a value such as 0 into a variable in this way is called assigning

- "0" is the value stored in the variable
- "=" is the assignment operator. It means "put the value into the variable".

الإخراج إلى وحدة التحكم

الإخراج إلى وحدة التحكم يعني عرض المعلومات على الشاشة داخل وحدة التحكم، مثال :

output to console

Output to console means showing information on the screen inside the console, Example :

```
main.js
let apple;
apple = 0;
console.log(apple);
```

```
Output
0
```

١- ما هو let apple;
- يمكنك تسمية المتغيرات بنفسك.
- كلمة let هي إشارة لإنشاء متغير جديد
- كلمة apple هنا هي اسم المتغير
- إنشاء متغير جديد يسمى التصريح declaring في عالم البرمجة

1- what is let apple;
- You can name variables by yourself
- The "let" is a signal to create a new variable
- "apple" here is the variable name
- Creating a new variable is called "declaring" in the programming world

٢- ما هو fish = 0;
- الرقم 0 هو القيمة
- وضع قيمة مثل 0 داخل متغير بهذه الطريقة يُسمّى الإسناد
- القيمة الأولى للمتغير يمكن أن تكون أي قيمة رقمية

2- what is fish = 0;
- 0 is the value
- putting a value such as 0 into a variable in this way is called assigning
- the first value of a variable can be any numerical value

main.js

```javascript
let fish;
fish = 0;
fish = fish + 3;
fish = fish + 7;
console.log(fish);
console.log("fish");
```

Output

```
10
fish
```

- variables can be changed to a variety of values
- adding like +3 and +7 to the variable fish is called reassigning
- console.log used to output the value of the variable fish to the console
- variables are not characters you have to remove the double quotation marks in order to output it
- enclosing fish with double quotation marks will output the text fish not the value

- يمكن تغيير المتغيرات إلى مجموعة متنوعة من القيم
- إضافة قيم مثل +3 و +7 إلى المتغير fish يسمى إعادة الإسناد (reassigning)
- تستخدم console.log لإخراج قيمة المتغير fish إلى وحدة التحكم
- المتغيرات ليست نصوصًا، لذلك يجب إزالة علامتي الاقتباس المزدوجتين لإخراج قيمتها
- وضع fish بين علامتي اقتباس مزدوجتين سيؤدي إلى إخراج النص fish نفسه وليس قيمته

Chapter 5 : if statement

الفصل الخامس : عبارة if

The if statement

جملة if

1 what is the if statement ?

- The if statement is used to execute a block of code only if a specific condition is true
- Conditional branching simply means doing something under certain conditions, such as "if ~ then do"
- if the condition is false, the code block is skipped

١ ما هي جملة if ؟

- تستخدم جملة if لتنفيذ كتلة من الكود فقط إذا كان شرط معين صحيحًا
- التفرع الشرطي يعني ببساطة القيام بشيء ما تحت شروط معينة، مثل "إذا ~ فقم بـ"
- إذا كان الشرط خاطئًا، يتم تجاهل كتلة الكود

2 Conditional branching program

To write a Conditional branching program, you must follow these steps :

٢ برنامج التفرع الشرطي

لكتابة برنامج يعتمد على التفرع الشرطي، يجب عليك اتباع هذه الخطوات:

main.js

```
let grade = 100;
if (grade == 100)
{console.log("Excellent");}
```

Output

```
Excellent
```

- This program means if the value of the variable grade equal to 100, the console outputs Excellent
- When you see the word if at the beginning of the program, Its called Conditional branching or if statement
- when you use an if statement, the computer checks to see the specified condition is met, and if so, it performs the specified processing
- Conditional expressions are written in "( )" and processing is written in "{ }"
- The formula of if statment is :
  if (condition) { process }

- يعني هذا البرنامج أنه إذا كانت قيمة المتغير grade تساوي 100، فإن وحدة الإخراج تعرض كلمة Excellent
- عندما ترى كلمة if في بداية البرنامج، فإن ذلك يسمى التفرع الشرطي أو جملة if
- عند استخدام جملة if، يتحقق الكمبيوتر لمعرفة ما إذا كان الشرط المحدد مستوفى، وإذا كان كذلك، فإنه ينفذ المعالجة المحددة
- يتم كتابة التعابير الشرطية بين "( )" ويتم كتابة المعالجة بين "{ }"
- صيغة جملة if هي:
  if (الشرط) { العملية }

3 The difference between " = " and " == "

٣ الفرق بين " = " و " == "

1. **Assignment Operator (=) :**
   - Used to assign or reassign a value to a variable, Example : let x = 5;
2. **Equality Operator (==) :**
   - Used to compare two values
   - Returns true if the values are equal and false if they are different

١. **عامل الإسناد (=) :**
- يستخدم لإسناد أو إعادة إسناد قيمة لمتغير، مثال: let x = 5;

٢. **عامل المساواة (==) :**
- يستخدم لمقارنة قيمتين
- يرجع true إذا كانت القيم متساوية و false إذا كانت غير متساوية

4 Validation

٤ التحقق

main.js

```
let grade = 100;
console.log(grade == 100);
```

Output

```
true
```

- The result of the comparison will output true in-case they are the same and outputs false in-case they are different

- ستظهر نتيجة المقارنة القيمة الصحيحة إذا كانت متطابقة، وتُظهِر القيمة الخاطئة إذا كانت مختلفة

Chapter 6 : If else statement

الفصل السادس : عبارة if else

The if ... else statement

جملة if ... else

1 what is the if-else statement ?

١ ما هي جملة if-else ؟

- The if ... else statement provides an alternative path
- The code inside the else block runs only if the if condition is false

- توفر جملة if ... else مسارًا بديلًا
- يتم تنفيذ الكود داخل كتلة else فقط إذا كان شرط if خاطئًا

main.js

```
let number = 9;
if (number == 10) {
console.log("The number is 10.");}
else { console.log("The number is not 10.");}
```

Output

```
The number is not 10.
```

- The code inside the else block is executed when the if condition is false
- The formula of if statment is :
  if (condition) { process of true }
  else { process of false }

- يتم تنفيذ الكود داخل كتلة else عندما تكون شرط if خاطئًا
- صيغة جملة if هي:
  if (الشرط) { العملية عند التحقق }
  else { العملية عند عدم التحقق }

Operator (1)

المشغل (١)

1 Operator (>=)

١ المشغل (>=)

main.js

```
let age = 18;
if (age >= 18) { console.log("You are an adult.");}
else { console.log("You are a minor.");}
```

Output

```
You are an adult.
```

- Here, the condition age >= 18 is true (18 is equal to 18), so the else block is not executed

- هنا، الشرط age >= 18 صحيح (18 يساوي 18)، لذا لم يتم تنفيذ كتلة else

2 Operator (<=)

```javascript
let temperature = 30;
if (temperature <= 25) {
  console.log("The weather is cool.");}
else { console.log("The weather is warm.");}
```

In this example, the "<=" operator checks if the temperature is less than or equal to 25. If true, it prints "The weather is cool.", otherwise, it prints "The weather is warm."

٢ المشغل (<=)

```
The weather is warm.
```

في هذا المثال، يقوم المشغل "<=" بالتحقق مما إذا كانت درجة الحرارة أقل من أو تساوي 25. إذا كان الأمر كذلك، يتم طباعة "The weather is cool."، وإذا لم يكن، يتم طباعة "The weather is warm."

Chapter 6 : If else statement

الفصل السادس : عبارة if else

The if ... else if ... else statement

عبارة if ... else if ... else

1 what is the if - else if - else statement ?

- The if ... else if ... else statement is used to check multiple conditions in sequence
- The code block of the first true condition is executed
- The final else block is optional and runs if none of the conditions are true

١ ما هي جملة if - else if - else ؟

- تستخدم جملة if ... else if ... else للتحقق من شروط متعددة بالتسلسل
- يتم تنفيذ كتلة الكود الخاصة بالشرط الصحيح الأول
- كتلة else الاخيرة اختيارية ويتم تنفيذها إذا لم يكن أي من الشروط صحيحًا

main.js

```
let score = 75;
if (score >= 90) {
  console.log("Excellent");}
else if (score >= 70) {
  console.log("Good");}
else {console.log("Failed");}
```

Output

```
Good
```

- In this code, the first condition score >= 90 is false, the second condition score >=70 is true (75 is bigger than 70), so "Good" is printed
- The formula of if statment is :
  if (condition) { process }
  else if (condition) { process }
  else { process }

- في هذا الكود، الشرط الأول score >= 90 غير صحيح، الشرط الثاني score >= 70 صحيح (75 أكبر من 70)، لذا سيتم طباعة "Good"
- صيغة جملة if هي:
  if (الشرط) { العملية }
  else if (الشرط) { العملية }
  else { العملية }

Chapter 9 : Logical Operators
الفصل التاسع : المعاملات المنطقية

Logical Operators
المعاملات المنطقية

1 Logical Operators

- Logical operators allows you to combine multiple conditions in an if statement
- They help make more complex decisions based on several criteria

١ المعاملات المنطقية

- تسمح لك المعاملات المنطقية بدمج شروط متعددة في جملة if
- تساعد في اتخاذ قرارات أكثر تعقيدًا بناءً على معايير متعددة

2 The logical AND operator (&&)

- "&&" operator is called Ampersand
- The "&&" operator checks if all connected conditions are true
- The entire expression is true only if every single condition is met

٢ معامل AND المنطقي (&&)

- المعامل "&&" يسمى Ampersand
- يتحقق المعامل "&&" مما إذا كانت جميع الشروط المتصلة صحيحة
- يكون التعبير بأكمله صحيحًا فقط إذا تم استيفاء كل الشروط

main.js

```
let people = 3;
let bill = 3500;
if (people >= 2 && bill >= 3000)
 { console.log("20% OFF");}
```

Output

```
20% OFF
```

- Discount is applied because both conditions are met : people are more than 2 AND bill is more than 3000

- تم تطبيق الخصم لأن كِلا الشرطين قد تحققا : عدد الاشخاص اكثر من 2 و الفاتورة اكثر من 3000



Mr. Sief Tamer - أ. سيف تامر
شرح الدرس كامل على اليوتيوب

3 The logical OR operator (||)

- "||" operator is called Vertical line / Pipe
- The " || " operator checks if at least one of the connected conditions is true
- The entire expression is true if any condition is met

```
main.js
let age = 10;
if (age <= 12 || age >= 65) {
  console.log("30% OFF");}
```

- Discount is applied because one condition is met : age is less than 12

٣ معامل OR المنطقي (||)

- المعامل "||" يسمى Vertical line أو Pipe
- يتحقق المعامل " || " مما إذا كان شرط واحد على الأقل من الشروط المتصلة صحيحًا
- يكون التعبير بأكمله صحيحًا إذا تم استيفاء أي شرط

```
Output
30% OFF
```

- تم تطبيق الخصم لأن شرطًا واحدًا قد تحقق : العمر اقل من 12

الفصل العاشر : العملية التكرارية

Chapter 10 : Iterative operation

سطر جديد

newline

١ استخدام فواصل الأسطر لسهولة القراءة

يمكن أن يكون من الصعب قراءة الأسطر الطويلة من الكود مثل عبارات if المعقدة التي تحتوي على شروط متعددة. يمكن استخدام سطر جديد (فاصل أسطر) لتقسيم الكود إلى عدة أسطر، مما يجعله أكثر وضوحًا وأسهل في الفهم. غالبًا ما يتم ذلك بعد قوس الفتح "{".
سيظل البرنامج يعمل بشكل صحيح

1 using line breaks for readability

Long lines of code such as complex if statements with multiple conditions, can be difficult to read. a newline (line break) can be used to split the code into multiple lines, making it cleaner and easier to understand. This often done after an opening curly brace "{".
The program will still execute correctly

```
main.js
let numberOfPeople = 2;
let fee = 3500;
if (numberOfPeople >= 2 && fee >= 3000) {console.log("Free mug")}
```



```
main.js
let numberOfPeople = 2;
let fee = 3500;
if (numberOfPeople >= 2 && fee >= 3000) {
  console.log("Free mug")
}
```

- عند البدء في سطر جديد، سيتم إضافة مسافة (indentation) تلقائيًا، مما يجعل البرنامج أسهل في القراءة

- When you start a new line, space (indentation) will be added automatically, it makes the program easier to read

العملية التكرارية

Iterative Process

قم بإخراج كلمة "Silk" عشر مرات :

Output "Silk" ten times :

ممارسة سيئة

Bad Practice

كتابة console.log("Silk"); في عشرة أسطر

Typing console.log("Silk"); in ten lines

main.js
```
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
console.log("Silk");
```

Output
```
Silk
Silk
Silk
Silk
Silk
Silk
Silk
Silk
Silk
Silk
```

١ for loop

1 for loop

- تستخدم حلقة for لتكرار كتلة من التعليمات البرمجية عددًا محددًا من المرات

- The for loop is used to repeat a block of code a specific number of times

main.js
```
for (let count = 0; count < 10; count = count + 1) {
  console.log("Silk")
}
```

- في هذا الكود، استخدمنا حلقة for لعرض الكلمة "Silk" عشر مرات.
- قمنا بتهيئة عداد باسم "count" بقيمة 0، ووحدنا الشرط ليكون أقل من 10 وزدناه بمقدار 1 في كل تكرار.
- صيغة حلقة for هي:
  for (التهيئة; الشرط; التحديث) {
  الكود
  }

- In this code, used for loop to output the word "Silk" ten times
- We initialized a counter named "count" with a value of 0, set the condition of it to be less than 10 and increment it by 1 in each iteration
- The formula of for loop is :
  for (initialization; Condition; Update) {
  code
  }

Increment

الزيادة

1 The increment operator (++)

- The increment operator ++ is a shorthand way to increase the value of a variable by 1
- instead of writing count = count + 1, you can simply write count++

١ عامل الزيادة (++)

- المعامل ++ هو طريقة مختصرة لزيادة قيمة متغير بمقدار 1
- بدلاً من كتابة count = count + 1، يمكنك ببساطة كتابة count++

main.js

```
let count = 0;
count = count + 1;
```



main.js

```
let count = 0;
count++;
```

2 Increment in for loop

- The ++ operator makes the Update part of a for loop cleaner and more coincise
- It is the most common way to increment the loop counter

٢ الزيادة في حلقة loop

- عامل ++ يجعل جزء التحديث في حلقة For أكثر نظافة وإيجازًا
- هذه هي الطريقة الأكثر شيوعًا لزيادة عداد الحلقة

main.js

```
for (let i = 0; i < 10; i++) {
  console.log("Silk")
}
```

Output

```
1   Silk
    ...
10  Silk
```

Chapter 12 : Functions

الفصل الثاني عشر : الدوال

Functions

الدوال

1 The Concept of a Functions

A function is a block of code designed to perform a particular task. Instead or rewriting the same code multiple times, you define it once and 'call' it whenever needed.

١ مفهوم الدالة

الدالة هي مجموعة من الأوامر التي تنفذ مهمة محددة. بدلًا من كتابة نفس الكود عدة مرات, يتم تعريفها مرة واحدة و استدعاؤها عند الحاجة

2 Built-in Functions

JavaScript include pre-written functions to perform common tasks

٢ الدوال المدمجة

تتضمن جافا سكريبت دوال مكتوبة مسبقًا لأداء مهام شائعة

Math.round( )

- Math.round( ) rounds a number to the nearest integar
- if the fraction is 0.5 or greater, it rounds up. Otherwise, it rounds down

Math.round( )

- تقوم دالة Math.round( ) بتقريب الرقم الى اقرب عدد صحيح
- إذا كان الكسر 0.5 او اكبر يتم التقريب للأعلى. أما إذا كانت أقل من ذلك, يتم التقريب للأسفل

main.js
```
console.log(Math.round(9.6));
console.log(Math.round(10.3));
```

Output
```
10
10
```

Math.max( )

- Math.max( ) finds the highest value in a list of numbers

Math.max( )

- تقوم دالة Math.max( ) بإيجاد القيمة الأكبر في قائمة الأرقام

main.js
```
console.log(Math.max(5, 10, 50));
```

Output
```
50
```

Math.min( )

- Math.min( ) finds the lowest value in a list of numbers

main.js
```
console.log(Math.min(5, 10, 50));
```

Math.min( )

- تقوم دالة Math.min( ) بإيجاد القيمة الأصغر في قائمة الأرقام

Output
```
5
```

How to create a Function

كيفية إنشاء دالة

1 The Four parts of a Function

1. **Function keyword :**
   To create a custom function, start with the function keyword, followed by a unique name and parantheses '( )' for the inputs/arguments
2. **Input (Arguments) :**
   The data you pass to the function, enclosed in parantheses '( )'
3. **Process (Body) :**
   The code inside the curly braces '{ }' that performs the task
4. **Output (Return Value) :**
   The final result sent back by the return keyword

١ أجزاء الدالة الأربعة

١. **الكلمة المفتاحية Function :**
   لإنشاء دالة مخصصة، ابدأ بكلمة function المفتاحية، متبوعة باسم فريد وأقواس '( )' لتمرير المدخلات/الوسائط
٢. **المدخلات (الوسائط) :**
   البيانات التي تمررها إلى الدالة، موضوعة بين أقواس '( )'
٣. **العملية (جسم الدالة) :**
   الكود الموجود داخل الأقواس المعقوفة '{ }' الذي ينفذ المهمة
٤. **المخرجات (القيمة المرجعة) :**
   النتيجة النهائية التي يتم إرجاعها بواسطة كلمة return

main.js
```
function functionName(parameter1, parameter2) {
  // code
  return value;
}
```


أ. سيف تامر - Mr. Sief Tamer

شرح الدرس كامل على اليوتيوب

2 Defining a Custom Function

٢ تعريف دالة مخصصة

This function takes two numbers and returns the smaller one

تأخذ هذه الدالة رقمين وتعيد الرقم الأصغر منهما

main.js

```
function min2(num1, num2) {
  let result;
  if (num1 < num2) {
    result = num1;
  } else {
    result = num2;
  }
  return result;
}
```

main.js

```
console.log(min2(5, 10));
```

Output

```
5
```

Important rules :

- Functions name cannot contain dots
- JavaScript is case-sensitive, Example : Math is not math

قواعد هامة :

- لا يمكن أن تحتوي أسماء الدوال على نقاط
- لغة جافا سكريبت حساسة لحالة الأحرف، على سبيل المثال: Math ليست نفسها math

الفصل الثالث عشر : الدوال [٢]

Chapter 13 : Functions [2]

الدوال [٢]

Functions [2]

١ الدوال المدمجة

1 Built-in Functions

دوال مكتوبة مسبقًا في جافا سكريبت

pre-written functions in JavaScript

Math.floor( )

- تقوم دالة Math.floor( ) بتقريب الرقم للأسفل إلى أقرب عدد صحيح
- هي فعليًا تقوم بإزالة (اقتطاع) الجزء العشري من الرقم

- Math.floor( ) rounds a number down to the nearest integar
- it effectively removes (truncate) the decimal part of a number

main.js

```
console.log(Math.floor(7.7));
```

Output

```
7
```

Math.random( )

- تقوم دالة Math.random( ) بإرجاع رقمًا عشريًا بين 0 (مشمول) و 1 (غير مشمول)
- 0 ≥ الناتج > 1
- الناتج يكون مختلفًا في كل مرة يتم استدعاؤها

- Math.random( ) returns decimal number between 0 (included) and 1 (not included)
- 0 ≤ result < 1
- The output is different everytime you call it

main.js

```
console.log(Math.random());
console.log(Math.random());
console.log(Math.random());
```

Output

```
0.6954556490529853
0.7146757984219194
0.30124178989299355
```

- لزيادة مدى Math.random() يمكنك ضرب ناتج الدالة

- To increase the range of Math.random() you can multiply the function

main.js

```
console.log(Math.random() * 10);
console.log(Math.random() * 10);
console.log(Math.random() * 10);
```

Output

```
9.729950745559567
6.234692907086106
4.704047585289226
```

- You can combine more than one function together like Math.floor( ) and Math.random( )

- يمكنك دمج أكثر من دالة معًا مثل Math.floor( ) و Math.random( )

```
main.js
console.log(Math.floor(Math.random() * 10));
console.log(Math.floor(Math.random() * 10));
console.log(Math.floor(Math.random() * 10));
```

```
Output
6
5
8
```

- This code outputs a random number between 0 and 9

- يقوم هذا الكود بإخراج رقم عشوائي بين 0 و 9

Math.random( ) + if statement

عبارة Math.random( ) + if

- You can use Math.random( ) as a condition for if statment to get randomized output

- يمكنك استخدام الدالة Math.random() كشرط في عبارة if للحصول على نتائج عشوائية

```
main.js
if (Math.random() >= 0.5) {
  console.log("Lucky Today!");
} else {
  console.log("Unlucky Today");
}
```

```
Output
// randomized output between
Lucky Today!
Unlucky Today
```


أ. سيف تامر - Mr. Sief Tamer
شرح الدرس كامل على اليوتيوب

الفصل الخامس عشر : أنواع البيانات

Chapter 15 : Data Types

أنواع البيانات

Data Types

١ الدوال المدمجة

- يمكن للمتغيرات تخزين أنواع مختلفة من البيانات. يتم إسناد القيمة العددية مباشرة، بينما يجب أن تكون القيمة النصية محاطة بعلامتي اقتباس مزدوجتين.
- يتعامل الحاسوب مع (1000) كرقم لإجراء العمليات الحسابية، أما ("1000") فيتم التعامل معه كنص.

1 Character Assignment

- Variables can store different type of data. A numerical value is assigned directly, while a string (Text) value must be enclosed in double quotes
- The computer treats (1000) as a number for calculation, but ("1000") as a text.

٢ اختلاف انواع البيانات

- يتصرف العاملون بشكل مختلف بناءً على نوع البيانات. إضافة الأرقام تُجرى العمليات الحسابية، بينما إضافة السلاسل النصية تقوم بدمجها

2 Data Type Difference

- Operators behave differently based on the data type. Adding numbers performs arithmetics, while adding strings concatenates (join) them

main.js

```
console.log(1 + 1);
console.log("1" + "1");
```

Output

```
2
11
```

٣ تحديد انواع البيانات

- يستخدم العامل (typeof) لتحديد نوع البيانات لقيمة معينة.
- ويرجع هذا العامل سلسلة نصية تشير إلى نوع البيانات.

3 Identifying Data Types

- The (typeof) operator is used to determine the data type of a value
- It returns a string indicating the type

main.js

```
console.log(typeof 100);
console.log(typeof "100");
```

Output

```
number
string
```

4 Comparison Operators (== vs ===)

- == (equality) checks only the value
- === (strict equality) checks both the value and the data type

```
main.js
console.log(100 == "100");
console.log(100 === "100");
```

- Using === is generally recommended

٤ معاملات المقارنة (== ضد ===)

- == (المساواة) تتحقق من القيمة فقط
- === (المساواة الصارمة) تتحقق من كل من القيمة و نوع البيانات

```
Output
true
false
```

- يوصى عادةً باستخدام ===

Type Conversion

تحويل النوع

1 Converting to Number

- The Number( ) function is used to convert a string value into a numerical value
- If the string is not valid number, the result will be NaN (Not a Number)

```
main.js
console.log(Number("10"));
console.log(Number("Hello"));
```

١ التحويل إلى رقم

- تُستخدم الدالة Number( ) لتحويل القيمة النصية إلى قيمة رقمية
- إذا لم تكن السلسلة النصية رقمًا صالحًا، ستكون النتيجة NaN (ليست رقمًا).

```
Output
10
NaN
```

2 Converting to String

- The String( ) function is used to convert a numerical value into a string value

```
main.js
console.log(String(500));
```

٢ التحويل إلى سلسلة

- تُستخدم الدالة String( ) لتحويل القيمة الرقمية إلى قيمة نصية

```
Output
"500"
```

مجانا وحصريا

حمل الآن

المراجعة رقم (3)

الترم الاول





Chapter 1

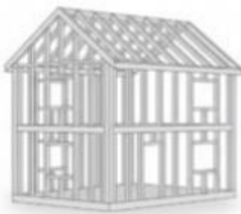
Programming

It means instructions provided to a computer , these instructions are called a "program"

programming allows you to create:



- There are more than 100 programming languages in the world



HTML
Build the framework
HTML



CSS
Create appearance
HTML + CSS



JavaScript
Create movement
HTML + CSS + JavaScript

- The black screen where you write your program is called:
  text editor or console.

Example:

- Write a program to output "Hello".

JavaScript
```
console.log("Hello");
```

NOTES

- Text should be added between " "
- To write a comment: start the sentence with //

Review

1. Select the correct program to output "Hello" to the console.

a) log.console("Hello");

b) console.log("Hello");

c) consolelog("Hello")

d) console.log(Hello);

2. **Create a program to output "Good morning" to the console.**

……………………………………………………………………………………………

3. Select the correct way to write your comment.

a) /Write your comments here./

b) //Write your comments here.

c) /Write your comments here.

d) Write your comments here.//

Chapter 2
Calculations and Strings

1 **To output *text*:**

write text with " " in *console.log( );*

2 **To output *calculation* results:**

write the calculation formula in brackets without " " in *console.log( );*

JavaScript

```
console.log("Hello World");
console.log("2+2");
console.log(2+2);
```

Console

```
Hello World
2+2
4
```

- Add: console.log(32 +48)
- Subtract: console.log(100 - 40)
- Multiply: console.log(5 * 9)
- Divid: console.log(10/2)

- To connect characters, numbers and calculations in a one-line program, you can use "+" to join each part:

JavaScript

```
console.log("Set of 2 coffes "+(250*2)+"coins");
```

JavaScript

```
console.log("100 + 200" + “is” + (100 + 200));
```

Result console

```
100 + 200  is  300
```

Revision

1. **Select the correct program to output the calculation results of "50+100" to the console.**

a) console.log("50" + "100");
b) console.log(50" + "100);
c) console.log(50 + 100);
d) console.log("50 + 100");

2. **Select the correct symbol for the division used in the program.**

a) *
b) /
c) #
d) ÷

3. **Let the computer calculate "19+40" and output the calculation results to the console.**

....

4. **Let the computer calculate "7-2" and output the calculation results to the console.**

....

5. **Let the computer calculate "15 divided by 5" and output the calculation results to the console.**

....

6. **Select the correct program to output "Today is the 17th.7 days ago was the 10th." to the console.**

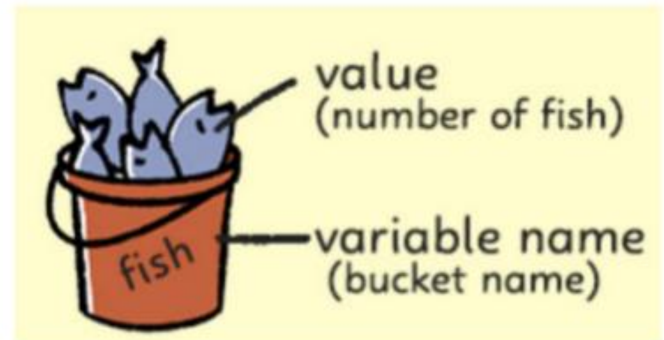
a) console.log("Today is the 17th.7 days ago was the + 17 - 7 + th.");
b) console.log("Today is the 17th.7 days ago was the " + "17 - 7" + "th.");
c) console.log("Today is the 17th.7 days ago was the " + (17 - 7) + "th.");
d) console.log("Today is the 17th.7 days ago was the (17 - 7)th.");

7. **Let the computer calculate "2*5" and output "The answer is 10." to the console.**

....

Chapter 3
Variables

- **Variable**: It is like a box for storing a value.
- Each variable has a value.
- The value of a variable can be changed.



Step 1: Declare
`let item1;`
→ **Step 2: Assign**
`item1 = 25;`
→ **Step 3: Use**
`console.log(item1);`



Declare the variable fish, assign 0, assign fish+8, assign fish+12, and create a program to output the value of the variable to the console.

Solution:

Program:

```
let fish;
fish = 0;
fish = fish + 8;
fish = fish + 12;
console.log(fish);
```

Output:

Sample: Output

```
20
```

Revision

1. Create a variable number, Assign 100 to the variable number. then output the value of number to the console.

   ………………………………………………………………………………………
   ………………………………………………………………………………………
   ………………………………………………………………………………………

2. In the previous program, (1) Assign the value of the variable number plus 10 (2) Output the value of variable number to the console

   ………………………………………………………………………………………
   ………………………………………………………………………………………

3. What will be the output of the following code ?

```
let a = 5;
a = a + 3;
console.log("The result is " + a);
```

   ………………………………………………………………………………………

Chapter 4

Mid-Term Exam

1. Select the correct program to output "Hello" to the console.

   a) console.log("Hello"); b) consolelog("Hello")

   c) console.logHello; d) log.console("Hello");

2. Output "Hello" to the console.

   ....

3. Create your program as follows:

   -Change "Leave this text." to a comment.

   -Execute the program and it will output "Make the first line a comment."

```
JavaScript
Leave this text.
console.log("Make the first line a comment.");
```

   ....

   ....

4. Select the correct program that will let the computer calculate "5+7" and output the calculation results to the console.

   a) console.log(5" + "7); b)console.log("5 + 7");

   c) console.log("5" + "7"); d) console.log(5 + 7);

5. Select the correct symbol for multiplication used in the program.

   a) / b) + c) - d) *

6. Let the computer calculate "3*5" and output "The answer is 15." to the console.

   ....

   ....

7. Select the correct declaration for the variable number.

   a) let = number; b) number let;

   c) number; d) let number;

8. Create your program as follows

   (1) Declare the variable num.

   (2) Assign 5 to num

   (3) Output the value of num to the console

   ....

   ....

Chapter 5
If statement

Case1: Test 1 condition

If (condition) {processing : This code is executed if condition is true}



Declare the variable science, Assign "100" to it, and if the variable science is "100", Output "Excellent".

Solution

JavaScript — Run Code ▶

```javascript
let science = 100;
if (science == 100){console.log("Excellent");}
```

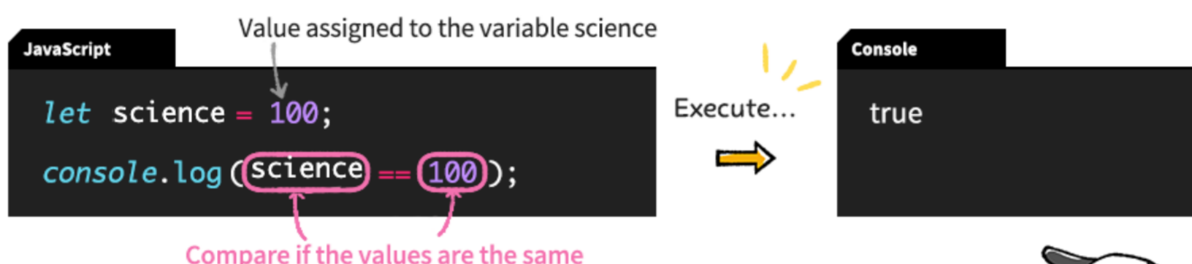
Console

```
Excellent
```



1. We use "=" : to assign value to a variable
2. We use "==": to compare two values.

You can use conditional expression without an if statement, The result will be (True or False)



Revision

1. Create your program as follows.
   (1) Declare the variable num and assign 5 to it.
   (2) Output the value of variable num to the console.
   However, the entire program should be written on two lines (the declaration and initial value assignment should be written on one line).
   ....
   ....
2. Select a program that says, "If ~, then ________."
   a) (conditional) if {processing}   b) (conditional){processing} if
   c) if {processing}(conditional)   d) if (conditional){processing}
3. Select the conditional expression "the value of the variable num is equal to 50".
   a) 50 >= num   b) num = 50
   c) num == 50   d) 50 = num
4. Create your program as follows
   (1) Declare the variable num and assign 100.
   (2) If variable num is equal to 100, output "Correct".
5. Please select the output result of the following program.

```
let num = 200;
console.log(num == 100);
```

   a) true   b) num
   c) 100   d) false

Comprehensive revision on chapters (1-5)

1. Which line has an error?

```
console.log(see you tomorrow);
console.log("see you tomorrow");
```

2. Write a JavaScript program to calculate 19 + 40 and output the result.
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………

3. Select the correct way to declare a variable:
   a) let = number; b) number let;
   c) number; d) let number;

4. Write a JavaScript program that calculates the total cost of three shopping items and gives a message based on the total amount.
   Instructions:
   1. Add the following comment at the top of your program :" *This program calculates total shopping cost"*
   2. Create three variables named item1, item2, and item3.
   3. Assign any numeric values to them (25, 15, 10).
   4. Calculate the total cost and store it in a variable named total.
   5. Output the price of each item and the total cost.
   6. Calculate and output the average price per item.
   7. Use an if statement:
      - If total is **50 or more**, output "Get a discount!".
      - Otherwise, output "Add more items to get a discount!".

   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………
   ……………………………………………………………………………………………………

Chapter 6
if-else statement

Case2

- It is used when you want to execute processing when the conditional expression is **true** or **false**, respectively

- If (condition) { A }
  else { B }



Comparison operators *Not all

1	A==B	A is the same as to B
2	A>B	A is greater than B
3	A<B	A is less than B
4	A>=B	A is greater than or equal to B
5	A<=B	A is less than or equal to B



Declare variable Math, Assign 80 to it.
Output "Excellent" if the variable Math is greater than or equal 90.
If not, the program outputs "You need to study hard ".

Solution

JavaScript

```
let math = 80;
if (math >= 90) {console.log("Excellent");}
else {console.log("You need to study hard");}
```

Run Code ▶

Console

```
You need to study hard
```

Review

1. Select a program that says if ~ then do A, otherwise do B.
   a) if (conditional){A}
      else{B}
   b) if (conditional) else{B}
      {A}
   c) else (conditional){A}
      if {B}
   d) else{B}
      if (conditional){A}
2. Create your program as follows.
   (1) Declare the variable num and assign 5 to it.
   (2) If variable num is the same as 10, output "Win".
   (3) Otherwise, output "Lose".
   ....
   ....
   ....
   ....
3. Select a program with the conditional expression "the value of the variable num is less than or equal to 80".
   a) num < 80
   b) num == 80
   c) num <= 80
   d) num > 80
4. Select a program with the condition "the value of the variable num is greater than 30".
   a) num < 30
   b) num == 30
   c) num <= 30
   d) num > 30
5. Create your program as follows
   (1) If the variable score is greater than or equal to 60, “Pass" is output; otherwise, “Fail" is output.
   Please do not delete the program that has already been written and input the conditional expression part.
   ....
   ....
   ....
   ....

Chapter 7
else if statements

Case3: Test multiple conditions

» If (condition 1) { A }
else if (condition 1) { B }
else if (condition 2) { C }
else if (condition 3) { D }



Create your program as follows
(1) Declare the variable "egg" and assign 5 to it.
(2) If the variable egg is 0, output "No stock".
(3) If not, and the variable egg is less than or equal to 3, output "Low stock".
(4) If not, output "In stock".

Solution

JavaScript

```javascript
let egg = 5;
if (egg == 0){console.log("No stock");}
else if (egg <= 3){console.log("Low stock");}
else {console.log("In stock");}
```

Console

```
In stock
```

» **Note:** A program is executed sequentially, starting from the top.

Review

1. Choose a program that says, "If ~, do A; if not, do B."

a) if (conditional){A}
   else if {B}

b) if (conditional){A}{B}
   else if (conditional)

c) if (conditional){A}
   else if (conditional){B}

d) if (conditional) (conditional)
   else if {A}{B}

2. select the output result of the following program.

```
let score = 62;
if (score == 100){console.log("Perfect score");}
else if (score >= 80){console.log("So close");}
else if (score >= 60){console.log("Pass");}
```

a) So close

b) Perfect score

b) Pass

d) No output

3. Create your program as follows
   (1) Declare the variable "egg" and assign 5 to it.
   (2) If the variable egg is 0, output "No stock".
   (3) If not, and the variable egg is less than or equal to 3, output "Low stock".
   (4) If not, output "In stock".

Chapter 8
Revision (Mid-term)

1. Create your program as follows
   (1) Declare the variable num and assign 7.
   (2) If variable num is equal to 7, output "Correct".

   ....

   ....

2. Please select the output result of the following program.

```
let num = 10;
console.log(num == 10);
```

a) 10 b) true
c) false d) 100

3. Select a program with the conditional expression "the value of the variable num is less than 20".

a) num >= 20 b) num <= 20
c) num == 20 d) num < 20

4. Create your program as follows.
   (1) Declare the variable num and assign 100 to it.
   (2) If variable num is equal to 100, output "correct". (3) If not, "miss" is output.

   ....

   ....

5. Please select the output result of the following program.

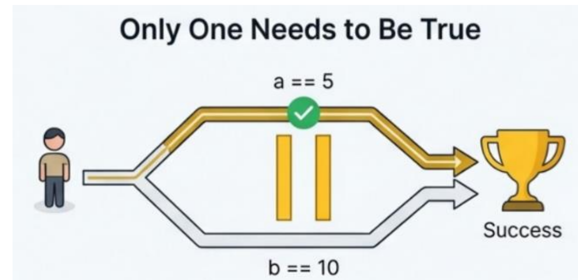
```
let score = 92;
if (score == 100){console.log("Perfect score");}
else if (score >= 80){console.log("So close");}
else if (score >= 60){console.log("Pass");}
```

a) So close b) Perfect score
c) No output d) Pass

Chapter 9
Logical operator

Symbol	Name	meaning	Example
&&	AND	Both conditions should be true	if (age > 18 && score >= 50){ }
||	OR	Only one condition should be true	if (age >= 18 || age < 18){ }





EX Create your program as follows.

(1) Declare the variable **numberOfPeople** and assign 2.
Declare the variable **fee** and assign 3500.
If the variable **numberOfPeople** is greater than or equal to 2 and the variable **fee** is greater than or equal to 3000, "Free mug" is output to the console.

EX Create your program as follows.

(1) Declare the variable **age** and assign 15 to it.
(2) If the variable **age** is less than or equal to 12 or greater than or equal to 65, "30% off" is output to the console.

Review

1) Select a program that means "A and B"

a) A && B

b) A // B

c) A & B

d) A || B

2) Select the program that means "A or B".

a) A && B

b) A // B

c) A & B

d) A || B

3) Please select the output result when executing the following program.

```
let age = 20;
if (age <= 18 || age >= 60){console.log("20% OFF");}
else{console.log("Regular Price");}
```

a) Error

b) No output

c) Regular Price

d) 20% OFF

4) **Please create a program as follows.**

If variable a is equal to 5 or b is equal to 10, output "Condition met".

```
let a = 5;
let b = 10;
if (                    ){console.log("Condition met");}
```

5) Please create a program as follows.

If variable a is greater than or equal to 2 and b is greater than or equal to 3, output "Condition met.

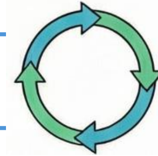
```
let a = 4;
let b = 3;
if (                    ){console.log("Condition met");}
```

Chapter 10

Iterative operation

For statement (Loops)

for (start; condition; update){ }



```
JavaScript
for (let count = 0; count < 10; count = count + 1 ) {
  console.log("silk");
}
```

Define variables — Conditional expressions — Update variables — Process



```
JavaScript
for (let count = 0; count < 10; count = count + 1 ) {
  console.log("silk");
}
```

Necessary — Necessary — Not necessary

";" (semicolon) means "a separator for the program," so it needs to be placed between parts of the program.

NOTES: - count = count + 1 can be written as **count++**

- in the programming world, in for statement ( ) , the variable is often named **i**



Create your program as follows.

(1) Output "campus" 10 times to the console.
The entire program should be 3 lines.
Variable name should be **count**.

NOTE: 1- Break a line after the left curly bracket " **{** " and before the right curly bracket "}".

2- Use <u>indentation</u> (space at line start) to make code easier to read



```
if (numberOfPeople >= 2 && fee >= 3000) {
  console.log("Free mug");
}
```

Revision

1) Select the first thing you would write in a program that represents "outputting the word "apple" to the console 10 times over.

a) let b) for

c) num d) if

2) Select the correct program to add 1 to variable i.

a) i+ b) i++

c) 1+1 d) i++1

3) Please select the correct answer for the program that repeats the specified process 10 times.

a) for (i<10; let i=0; i++) {

Specified process

}

b) for {Specified process} (

let i=0; i<10; i++

)

c) for (let i=0; i<10; i++) {

Specified process

}

d)

for (let i=0; i++; i<10) {

Specified process

}

4) Create your program less than or equal to
Output "Hello" to the console 5 times.
*Variable name is **i** and assign 0 at the beginning.
*Use the increment operator.

```
for (  ) {
 console.log("Hello");
}
```

....

....

5) Create your program less than or equal to
Output "Hello" to the console 7 times.
*Variable name is i and assign 0 at the beginning.
Use the increment operator.

....

....

Chapter 11

Mid Term Test

1) Select the program that means "A or B"

a) A && B
b) A < B
c) A == B
d) A || B

2) Please select the output result of running the program less than or equal to

```
Let num = 9;
if (num <= 3 || num >= 7) {console.log("A");}
else{console.log("B");}
```

a) B
b) 9
c) No output
d) A

3) Select the correct program to add 1 to variable i.

a) 1+1
b) i+
c) i++1
d) i++

4) Select the correct answer for the program that repeats the specified processing 10 times.

a) for { Specified process } (
    let i=0; i<10; i++
)

b) for (let i=0; i++; i<10) {
    Specified process
}

c) for (i<10; let i=0; i++) {
    Specified process
}

d) for (let i=0; i<10; i++) {
    Specified process
}

5) Please select the output result of running the program less than or equal to

```
for (let i = 0; i < 2; i++) {
 console.log("Hello");
};
```

a) Hello
Hello
Hello

b) No output

c) Hello

d) Hello
Hello

6) Create your program less than or equal to

(1) Output "Good morning" 10 times to the console.

*Variable name is **i** and assign 0 at the beginning.

Use the increment operator.

....

....

....

....

7) Create your program less than or equal to

If variable a is greater than or equal to 2 and **b** is greater than or equal to 3, output "Condition met.

*Only the conditional expression should be entered.

JavaScript

```javascript
let a = 4;
let b = 3;
if (________________) {
console.log("Condition met");
}
```

8) Create your program less than or equal to

(1) Output "Hello" three times to the console.

*Variable name is i and assign 0 at the beginning.

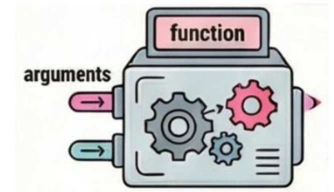
Use the increment operator.

JavaScript

```javascript
_____ (______________________________) {
 console.log("Hello");
}
```

Chapter 12

Functions (1)



- **Functions:** allow you to perform a predefined process quickly.
- Functions are named according to the processing they can perform.
- A function takes some data(**arguments**), performs a specified processing, and returns **the result**.

Types of Functions

1- Math.round(9.6)

- The function round off the value.

2- Math.max(num1,num 2, ...)

- The function tells the maximum value

3- Math.min(num1,num 2, ...)

- The function tells the minimum value

Problem1: Create a program to Round "9.7" using the function **Math.round** and output to the console

Answer:

JavaScript

```
console.log(Math.round(9.7));
```

NOTE:

1-The value given to the function is called the "argument" (such as 9.8, 10.3)

2-The value returned as a result of the calculation is called the "return value" (such as 10)

NOTE:

1. M in Math.max( ), Math.min( ),.... Should be in capital letter.
2. you cannot put "." (dot) in variable and function names.

How to create your own function ?

- **To create a function:**

function function_Name (argument) { process }

The program for "Mr. 3 Split"

```
function  split3(total)  {
let result = total / 3;
return result;
}
```

- ← Decide the names for the function and its arguments
- ← The actions (process) we want "Mr. 3 Split" to perform
- ← Return the result value



Problem1:

Create a function with the function name **split5** and the argument name **total**

Declare the variable result and assign the calculation formula "total / 5"

Pass the argument "12,500" to the function **split5** and output to the console

Answer:

JavaScript

```javascript
function split5(total) {
  let result = total / 5;
  return result;
}
console.log(split5(12500));
```

Problem2:

JavaScript

```javascript
function min2(num1,num2) {
  let result;
  if (num1 < num2) {
    result = num1;
  } else {
    result = num2;
  }
  return result;
}
```

This means we are

- creating a function that works like min2(num1, num2).

Revision

1. Select the correct program to create a function whose function name is **calc** and argument name is **num**.

a) (num)|calc function {
processing
}

b) function (num) calc {
processing
}

c) (num)function calc{
processing
}

d) function calc(num) {
processing
}

2. Select the correct function to round off.

a) Math.roundoff(1.8)

b) round(1.8)

c) Math.round(1.8)

d) round.Math(1.8)

3. Please select the output results of the following program that you have executed.

```
console.log(Math.max(20,40,60));
```

a) 120

b) 60

c) 20

d) 40

4. Please select the correct answer to be displayed in the console as a result of executing the following program:

```
function calc(num) {
  let result = num * 3;
  return result;
}
console.log(calc(5));
```

a) 15

b) 0

c) 3

d) 5

5. Please add a program that calls the function **calc**. Please put "3" as an argument.

JavaScript

```
function calc(num) {
  let result = num * 5;
  return result;
}
console.log(        );
```

6. Please add a program that returns the value of the variable result to the described program.

JavaScript

```javascript
function calc(num) {
  let result = num * 7;

}
console.log(calc(10));
```

Chapter 13

Functions (2)

4-Math.floor( )

it is a function that truncates decimals (rounds a number down) to the nearest integer!

EX *Math.floor(5.5)*

Problem 1:

Create your program as follows.

(1) Pass the argument 7.8 to the function Math.floor and output to the console

(2) Pass the argument 7.8 to the function Math.round and output to the console

JavaScript

```
console.log(Math.floor(7.8));
console.log(Math.round(7.8));
```



Console

```
7
8
```

5- Math.random()

It is a function that returns a random "numerical value greater than or equal to 0".



JavaScript

```
if (Math.random() >= 0.5) {
  console.log("Lucky today");
} else {
  console.log("Unlucky today");
}
```

NOTE: *Functions can be combined with other functions.*

JavaScript

```
console.log(Math.floor (Math.random() * 10));
```

Problem 2:

Output the minimum values of "5" and "50" to the console using the function Math.min.

Answer:

JavaScript

```
console.log(Math.min(5,50));
```

Problem 3:

Create a function that adds two numerical values.

(1) Create a function with the function name sum and the argument names num1 and num2

(2) Assign "num1 + num2" to the variable result

(3) Return the variable result

(4) Pass the arguments **10** and **20** to the function **sum** and call it to output to the console

Answer:

JavaScript

```
function sum(num1, num2) {
  let result = num1 + num2;
  return result;
}
console.log(sum(10,20));
```

```
Math.round(9.7)   : 10
Math.floor(7.8)   : 7
Math.max(10, 50)  : 50
Math.min(10, 50)  : 10
Math.random()     : 0-1
```

Revision

1. Select the correct program to output random numerical values.
   a) console.log(Math.min());
   b) console.log(random());
   c) console.log(Math.max());
   d) console.log(Math.random());

2. Select the correct function to truncate decimals.
   a) console.log(Math.round(10.5));
   b) console.log(Math.floor(10.5));
   c) console.log(Math.max(10.5));
   d) console.log(Math.min(10.5));

3. Select the output results of the following program.

```
console.log(Math.floor(7.7));
```

   a) No output
   b) 7
   c) 6
   d) 8

4. Enter the program as follows.
   (1) Use the function to round off "5.5" and output to the console.
   (2) Use the function to round down “8.8" to the nearest whole number and output to the console.
   Use the functions "Math.floor" and "Math.round".

   ....

   ....

5. Create a program that outputs “Excellent luck" if the random numerical value is greater than or equal to 0.5, and "Bad luck" otherwise.

   Use the function "Math.random()" and enter the conditional expression "if the random value is greater than or equal to 0.5".

JavaScript

```
if (                    ) {
  console.log("Excellent luck");
} else {
  console.log("Bad luck");
}
```

6. Create a program that returns a random integer as follows.
   (1) Use the function "Math.random" to return a random numerical value.
   (2) Multiply the randomly returned numerical value by 10.
   (3) Use the function "Math.floor" to return an integer.
   *The program is one line.

   ....

   ....

Chapter 14

Mid-term Test

1. Select the correct function to round off.

   a) round(1.8)
   b) Math.roundoff(1.8)
   c) Math.round(1.8)
   d) round.Math(1.8)

2. Select the output result of running the following program.

```
console.log(Math.max(10,50,100));
```

   a) 100
   b) 50
   c) 160
   d) 10

3. Select the correct program to output random numerical values.

   a) console.log(Math.min());
   b) console.log(Math.random());
   c) console.log(random());
   d) console.log(Math.max());

4. Select the output results of the following program.

```
console.log(Math.floor(12.8));
```

   a) 13
   b) 12
   c) No output
   d) 11

5. Create a program that returns a random integer as follows.
   (1) Use the function **Math.random** to return a random numerical value.
   (2) Multiply the randomly returned numerical value by 10.
   (3) Use the function **Math.floor** to return an integer.
   *The program is one line.

   ....

   ....

6. Select the correct program to create a function whose function name is **sum** and argument name is **num**.

a) function sum(num) {
    processing
}

b) (num)sum function {
    processing
}

c) (num)function sum{
    processing
}

d) function (num) sum {
    processing
}

7. Add a program that returns the value of the variable **result** to the described program.

JavaScript

```
function calc(num) {
  let result = num * 7;
  
}
console.log(calc(10));
```

8. Add a program that calls the function **calc**.
   Enter **7** as an argument

JavaScript

```
function calc(num) {
  let result = num * 5;
  return result;
}
console.log(          );
```

Chapter 15

Types of Variables

- **Number** — Such as: 5-10- 20
- **String** — Such as: "Welcome"



JavaScript
```javascript
console.log("1+1");
console.log(1+1);
```

- **To output the type of variables:**

JavaScript
```javascript
console.log(typeof 100);
```

JavaScript
```javascript
console.log(typeof "Hello");
```

- **To compare two values:**
  - **== :** It return "**True**" ,If only the **values** are the same
  - **===:** It return "**True**" ,If the **values** and the **type** are the same

```javascript
console.log(1 == "1");   // → true
console.log(1 === "1");  // → false
```

- **To change the type of a variable:**

1. console.log(Number("100")); **>>>>>Output>>>>>>>** **100** (Type: **number**)
2. console.log(String(100)); **>>>>> Output >>>>>>** **100** (Type: **string**)
3. console.log(Number("Hello")); **>>>>> Output >>>>>>** **NaN** (**Not a Number**)

Review

1. Select the correct data type below.

```
console.log("50+30");
```

   a) No particular data type
   b) Numerical value
   c) String
   d) None of them

2. Select the correct program that outputs "string".

   a) console.log(string "1");
   b) console.log(typeof "1");
   c) console.log(Number "1");
   d) console.log(typeof 1);

3. Select the output results of the following program.

   a) numerical value
   b) true
   c) false
   d) string

4. Select the output results of the following program.

   a) number
   b) true
   c) false
   d) NaN

5. Create your program as follows.

   (1) Determine the data type of "7" in the computer and output to the console.

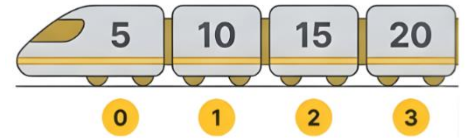
   ....

   ....

Chapter 16

Array

- **An array**: It is a collection of **multiple values** that can be added together.
- **Values** are separated by **,** in [ ]



JavaScript
```
let  variableName= [value1, value2, value3, ...];
```

Variable
```
let temp1 = 36;
let temp2 = 36;
let temp3 = 26;
let temp4 = 27;
let temp5 = 28;
let temp6 = 33;
let temp7 = 36;
```

Writing with Arrays
```
let tempList = [36, 36, 26, 27, 28, 33, 36];
```

- Each value in the array has location index **starts** from 0 .

```
  0    1    2    3    4    5    6
let tempList = [36, 36, 26, 27, 28, 33, 36];
  Day1 Day2 Day3 Day4 Day5 Day6 Day7
```

The temperature on the 5th means...



- **To retrieve the value of an array:**

variable name[location number] >>>>>> tempList [4]

- **To see all values of an array:**

Console.log(tempList);

JavaScript
```
let tempList = [36, 36, 26, 27, 28, 33, 36];
let total = 0;
for(let i = 0; i < 7; i++){
  total = total + tempList[i];
}
```

- **To calculate the length of an array(number of values):**

console.log(tempList.length);

Revision

1- Select the output result of executing the following program.

```
let num = [5, 10, 15, 20, 25];
console.log(num[2]);
```

a) 20

b) 2

c) 10

d) 15

5- Output all the values in the array to the console.

*Do not enter values directly.

*Do not delete the written program, but add the program in console.log.

JavaScript

```
let num = [5, 10, 15, 20, 25];
console.log(      );
```

6- Select the output result of executing the following program.

```
let num = [1, 2, 3];
let sum = 0;
for (let i = 0; i < 3; i++) {
  sum = sum + num[i];
}
console.log(sum);
```

a) 3

b) 6

c) 1

d) 2

7- Output the length of the array to the console.

*Do not enter the numerical value of the length directly.

JavaScript

```
let num = [5, 10, 15, 20, 25];
console.log(            );
```

Chapter 17

Midterm Test

1. Select the correct data type below.

```
console.log("11");
```

a) string
b) No particular type
c) Numerical value
d) boolean

2. Select the correct program that outputs "string".

a) console.log(String "50");
b) console.log(typeof "50");
c) console.log(Number "50");
d) console.log(typeof 50);

3. Select the output results of the following program.

```
console.log(200 === 20);
```

a) true
b) string
c) number
d) false

4. Select the output results of the following program.

```
console.log(Number("value"));
```

a) true
b) NaN
c) number
d) false

5. Create your program as follows.

(1) Computer determines the data type **100** and outputs it to the console

....

....

6. Put the following values in the array.

10, 50, 100, 150

*Nothing will be output when executed.

....

....

7. 7-Select the output result of executing the following program.

```
let num = [5, 10, 15, 20, 25];

console.log(num[3]);
```

a) 10 b) 20

c) 5 d) 15

8. Output all the values in the array to the console.
   *Do not enter values directly.
   *Do not delete the written program, but add the program in console.log.

JavaScript

```
let num = [2, 4, 6, 8, 10];
console.log(      );
```

9. Select the output result of executing the following program.

```
let num = [2, 4, 6];
let sum = 0;
for (let i = 0; i < 3; i++) {
  sum = sum + num[i];
}
 console.log(sum);
```

a) 12 b) 2

c) 6 d) 4

10. Output the length of the array to the console.
*Do not enter the numerical value of the length directly

JavaScript

```
let num = [2, 4, 6, 8, 10];
console.log( );
```

....

....

حمل الآن

مجانا وحصريا

المراجعة رقم (4)

الترم الاول





أوامر الطباعة والتعليقات (Console & Comments)	
// Comment	لترك ملاحظة في الكود أو إيقاف سطر
console.log();	أمر طباعة الكود بشكل عام
console.log("Text");	لطباعة نص (نستخدم علامتي اقتباس)
console.log(1 + 1);	لطباعة القيم العددية (لا تحتاج علامات اقتباس)
console.log(variable);	لطباعة قيمة متغير (لا يحتاج علامات اقتباس)

كل أكواد المنصة



abwaab
ابواب

المتغيرات والعمليات الحسابية (Variables & Math Operators)	
let x;	لإنشاء متغير
x = 5;	لإسناد قيمة للمتغير
العمليات الأساسية	
let sum = 1 + 2;	(+) لجمع الأرقام أو ربط النصوص
let sub = 5 - 2;	(-) لطرح الصيغ الحسابية
let mul = 5 * 2;	(*) لضرب الصيغ الحسابية
let div = 10 / 2;	(/) لقسمة الصيغ الحسابية



Mr.
Ahmed Hamed

abwaab
ابـواب

عوامل الزيادة والنقصان (تستخدم عادة في جملة For)	
(++) زيادة 1 إلى القيمة	i++;
(--) طرح 1 من القيمة	i--;



abwaab
ابواب

عمليات المقارنة (Comparison Operators)	
x == y;	للمقارنة بين قيمة متغير وقيمة محددة
x === y;	للمقارنة في القيمة ونوع البيانات معًا
<>	
A < B;	A أصغر من B
A > B;	B أصغر من A
A <= B;	A أصغر من أو يساوي B
A >= B;	A أكبر من أو يساوي B



abwaab
ابواب

الشروط (Conditions)	
جملة الشرط: إذا حدث... فافعل...	if (condition) { process }
إذا أردت إضافة تعبير شرطي آخر	else if (other Condition) { process }
إذا لم تحدث الشروط السابقة	else { process }



الروابط المنطقية (Logic)	
و (And)	&&
أو (Or)	||



abwaab
ابواب

(Loops & Functions) التكرار والدوال	
for (initialization; condition; increment) { code block }	لصناعة جملة تكرارية
function functionName(argumentName) { {process} }	إنشاء دالة (Function)



Mr.
Ahmed Hamed

abwaab
ابواب

(Math Object)	
Math.round(x);	الدالة التي تقوم بحساب التقريب
Math.max(x, y);	الدالة التي تخبرنا بالقيمة العظمى
Math.min(x, y);	الدالة التي تخبرنا بالقيمة الصغرى
Math.floor(x);	الدالة التي تزيل العدد العشري (تقريب للأدنى)
Math.random();	الدالة التي ترجع قيمة عشوائية



abwaab
ابواب

أنواع البيانات والتحويل (Data Types & Conversion)	
لتحديد نوع البيانات	typeof variable;
للتحويل من نص إلى قيمة عددية	Number("String");
للتحويل من قيمة عددية إلى نص	String(Number);



المصفوفات (Arrays)	
إسناد مصفوفة إلى متغير	let variable = [item1, item2, item3];
لطباعة إحدى أرقام المصفوفة // ملاحظة مهمة: نبدأ العد في المصفوفة من 0	console.log(variableName[0]);
للتحقق من طول المصفوفة	console.log(variableName.length);



المراجعة رقم (5)

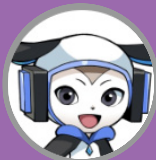
الترم الاول





+100 سؤال +100 سؤال +100 سؤال +100 سؤال





أسئلة المنصة



1 اختر الإجابة الصحيحة من بين الإختيارات التالية :



1. اختر البرنامج الصحيح لعرض «Hello» في وحدة التحكم.
   - أ) consolelog(“Hello”)
   - ب) console.log(“Hello”);
   - ج) log.console(“Hello”);
   - د) console.logHello;

2. اعرض «Good morning» في وحدة التحكم.
   - أ) log.console(“Good morning”);
   - ب) console.log(“Good morning”);
   - ج) log.console(Good morning);
   - د) console.log(“Good Morning”);

3. يرجى اختيار الطريقة الصحيحة لكتابة التعليق.
   - أ) //Write your comments here.
   - ب) Write your comments here.//
   - ج) /Write your comments here./
   - د) /Write your comments here.

4. اختر البرنامج الصحيح لعرض نتيجة حساب 50+100 في وحدة التحكم.
   - أ) console.log(50” + “100);
   - ب) console.log(“50 + 100”);
   - ج) console.log(50 + 100);
   - د) console.log(“50” + “100”);

5. اختر الرمز الصحيح للقسمة المستخدم في البرنامج.
   - أ) (÷)
   - ب) (+)
   - ج) (//)
   - د) (/)





6 اجعل الكمبيوتر يحسب 19+40 ويعرض نتيجة الحساب في وحدة التحكم.



أ console.log("19+40");
ب console.log(19+40);
ج console.log(19"40");
د log.console(19+40);

7 اختر البرنامج الصحيح لعرض «Today is the 17th. 7 days ago was the 10th.» في وحدة التحكم.

أ console.log("Today is the 17th.7 days ago was the (17 - 7)th.");
ب console.log("Today is the 17th.7 days ago was the " + (17 - 7) + "th.");
ج console.log("Today is the 17th.7 days ago was the + 17 - 7 + th.");
د console.log("Today is the 17th.7 days ago was the " + "17 - 7" + "th.");

8 اجعل الكمبيوتر يحسب 5*2 ويعرض «The answer is 10.» في وحدة التحكم.

أ console.log("The answer is 10.");
ب console.log("The answer is " +(2 * 5)+".");
ج console.log("The answer is 2 * 5");
د console.log("The answer is " +2 * 5+.);

9 اختر البرنامج الصحيح لإعلان المتغير score.



أ let score;
ب score let;
ج score;
د score =;

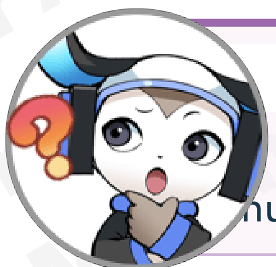
10 اختر البرنامج الصحيح لعرض قيمة المتغير score في وحدة التحكم.

أ console.log = score;
ب console.log score;
ج console.log(score);
د console.log("score");

11 تم الاعلان عن المتغير number أسند له القيمة 100 .

أ number ="100";
ب number100;
ج number =(100);
د number =100;

12 أسند قيمة المتغير number مضافاً إليها 10 .

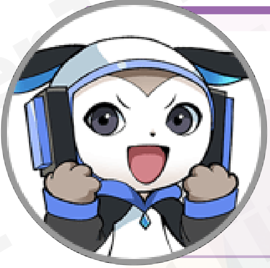


أ number + 10;
ب number = 100 + 10;
ج number = number + 10;
د number = number / 10;



13 اختر برنامجاً يقول: "إذا التعبير الشرطي إذن المعالجة".

أ if (conditional){processing}
ب (conditional) if {processing}
ج (conditional){processing} if
د if {processing}(conditional)



14 اختر عبارة الشرط "قيمة المتغير num تساوي 50".

أ num == 50
ب 50 >= num
ج 50 = num
د num = 50

15 يرجى اختيار نتيجة إخراج البرنامج التالي: let num = 200; console.log(num == 100);

أ num
ب true
ج 100
د false

16 اختر برنامجاً يقول إذا كان التعبير ~ فافعل A، وإلا فافعل B.

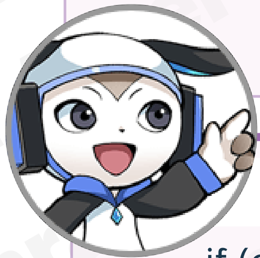
أ else (conditional){A} if {B}
ب if (conditional) else {B} {A}
ج if (conditional){A} else{B}
د else{B} if (conditional){A}

17 اختر برنامجاً يحتوي على التعبير الشرطي «قيمة المتغيّر num أصغر من أو تساوي 80».

أ num > 80
ب num <= 80
ج num < 80
د num == 80

18 اختر برنامجاً يحتوي على الشرط "قيمة المتغير num أكبر من 30".

أ num > 30
ب num < 30
ج num >= 30
د num == 30



19 اختر برنامجاً يقول "If ~, do A; if not, do B."

أ if (conditional){A}{B}else if (conditional)
ب if (conditional){A} else if {B}
ج if (conditional){A}else if (conditional){B}
د if (conditional) (conditional)else if {A}{B}



20 اختر نتيجة المخرجات للبرنامج التالي.



```
let score = 62;
if (score == 100){console.log("Perfect score");}
else if (score >= 80){console.log("So close");}
else if (score >= 60){console.log("Pass");}
```

أ Perfect score
ب Pass
ج No output
د So close

21 اختر نتيجة الإخراج للبرنامج التالي: let num = 10; console.log(num == 10);

أ 100
ب true
ج 10
د false

22 اختر نتيجة المخرجات للبرنامج التالي.

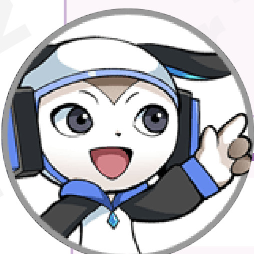


```
let score = 92;
if (score == 100){console.log("Perfect score");}
else if (score >= 80){console.log("So close");}
else if (score >= 60){console.log("Pass");}
```

أ Perfect score
ب Pass
ج No output
د So close

23 اختر برنامجاً يعني "A and B".

أ A // B
ب A || B
ج A & B
د A && B



24 اختر برنامجاً يعني "A or B".

أ A || B
ب A < B
ج A && B
د A == B



25 الرجاء اختيار نتيجة الإخراج عند تنفيذ البرنامج التالي.

```
let age = 20;
if (age <= 18 || age >= 60){console.log("20% OFF");}
else{console.log("Regular Price");}
```



أ السعر العادي
ب خصم 20%
ج خطأ
د لا يوجد إخراج

26 اختر أول شيء ستكتبه في برنامج يمثل "إخراج الكلمة apple إلى وحدة التحكم 10 مرات متتالية".

أ let
ب if
ج for
د num

27 اختر البرنامج الصحيح لإضافة 1 إلى المتغير i .

أ 1+1
ب i+
ج i++1
د i++

28 يرجى اختيار الإجابة الصحيحة للبرنامج الذي يكرّر العملية المحدّدة 10 مرات.

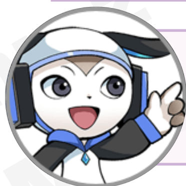
أ for (i<10; let i=0; i++) {Specified process}
ب ifor (let i=0; i++; i<10) {Specified process}
ج for (let i=0; i<10; i++) {Specified process}
د for {Specified process} (let i=0; i<10; i++)

29 الرجاء اختيار ناتج تشغيل البرنامج أقل من أو يساوي.

```
let num = 9;
if (num <= 3 || num >= 7) {console.log("A");}
else{console.log("B");}
```



أ A
ب B
ج 9
د لا يوجد إخراج



- البرنامج الذي يكرّر عملية محددة عددًا معينًا من المرات، يسمى "جملة for".وتسمى أيضًا "عملية تكرارية" أو "حلقة تكرار
- عامل الزيادة يُكتب "++"، ومعناه إضافة 1 إلى القيمة.



30 الرجاء اختيار ناتج تشغيل البرنامج أقل من أو يساوي.



```
for (let i = 0; i < 2; i++) {
 console.log("Hello");
}
```

أ Hello
ب No output
ج Hello Hello
د Hello Hello Hello

31 اختر البرنامج الصحيح لإنشاء دالة يكون اسمها calc واسم معاملها num.

أ function calc (num) {processing}
ب (num)calc function {processing}
ج function (num) calc {processing}
د (num)function calc {processing}

32 اختر الدالة الصحيحة للتقريب.

أ round.Math(1.8)
ب Math.round(1.8)
ج round(1.8)
د Math.roundoff(1.8)

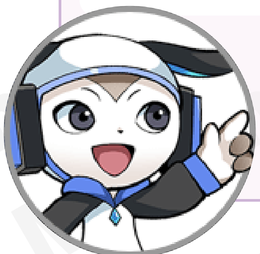
33 يرجى تحديد ناتج تشغيل البرنامج التالي: console.log(Math.max(20,40,60));

أ 20
ب 120
ج 60
د 40

34 الرجاء اختيار ناتج تشغيل البرنامج أقل من أو يساوي



```
function calc(num) {
  let result = num * 3;
  return result;
}
console.log(calc(5));
```



أ 5
ب 3
ج 0
د 15





35 اختر البرنامج الصحيح لإخراج القيم العشوائية.

أ `console.log(Math.max());` ب `console.log(Math.min());`

ج `console.log(random());` د `console.log(Math.random());`

36 اختر الدالة الصحيحة لقطع الكسور العشرية.

أ `console.log(Math.max(10.5));` ب `console.log(Math.min(10.5));`

ج `console.log(Math.floor(10.5));` د `console.log(Math.round(10.5));`

37 اختر ناتج تشغيل البرنامج التالي: `console.log(Math.floor(7.7));`

أ 7 ب 7.7

ج 8 د No output

38 اختر البرنامج الصحيح لإنشاء دالة يكون اسمها sum واسم الوسيط num.

أ `function sum(num) {processing}` ب `(num)function sum {processing}`

ج `(num)sum function {processing}` د `function(num) sum {processing}`

39 اختر الدالة الصحيحة لاستخراج القيمة العظمى من مجموعة قيم.

أ `console.log(Math.max(10,5));` ب `console.log(Math.min(10,5));`

ج `console.log(Math.floor(10.5));` د `console.log(Math.round(10.5));`

40 اختر الدالة الصحيحة لاستخراج القيمة الصغرى من مجموعة قيم.

أ `console.log(Math.max(20,50));` ب `console.log(Math.min(20,50));`

ج `console.log(Math.floor(20.5));` د `console.log(Math.round(20.5));`



- الدالة تأخذ بعض البيانات، تُجري معالجة محددة، ثم تُعيد النتيجة.
- القيمة التي تُمرَّر إلى الدالة تُسمى المعامل، والقيمة التي تُعاد تُسمى قيمة الإرجاع.





41 اختر نوع البيانات الصحيح للبرنامج التالي: console.log("50+30");

أ No particular data type
ب Numerical value
ج Boolean
د String

42 اختر البرنامج الصحيح الذي يُخرج **string**.

أ console.log(typeof "1");
ب console.log(string "1");
ج console.log(Number "1");
د console.log(typeof 1);

43 اختر ناتج تنفيذ البرنامج التالي: console.log(100 === 500);

أ string
ب false
ج numerical value
د true

44 اختر ناتج تنفيذ البرنامج التالي: console.log(Number("Hello"));

أ number
ب true
ج NaN
د false



45 اختر ناتج تنفيذ البرنامج التالي:

```
let num = [5, 10, 15, 20, 25];
console.log(num[2]);
```

أ 2
ب 20
ج 10
د 15

46 اختر ناتج تنفيذ البرنامج التالي:

```
let num = [1, 2, 3];
let sum = 0;
for (let i = 0; i < 3; i++) {
  sum = sum + num[i];
}
console.log(sum);
```

أ 1
ب 2
ج 3
د 6





47. HTML هي اختصار لـ .... .

أ) HyperText Markup Language
ب) HyperText Markup Link
ج) HighText Markup Language
د) HighText Machine Language

48. ما الاسم الذي يطلق على أوامر HTML .

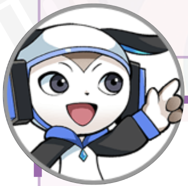
أ) Elements
ب) Attributes
ج) Properties
د) Tags

49. لكتابة أكبر عنوان في لغة HTML نستخدم الوسم .

أ) <h1> </h1>
ب) <h2> </h2>
ج)
د) <p> </p>

50. لكتابة عنوان فرعي في لغة HTML نستخدم الوسم .

أ) <h1> </h1>
ب) <h2> </h2>
ج)
د) <p> </p>



51. اختر الوسم المناسب لاستخدامه في كسر السطر في لغة HTML .

أ) b tag
ب) h5 tag
ج) br tag
د) p tag

52. اختر الوسم المناسب لإنشاء قائمة منقطة Unorderd List :

أ) ul tag
ب) ol tag
ج) br tag
د) p tag

53. اختر الوسم المناسب لإنشاء قائمة مرتبة Orderd List :

أ) ul tag
ب) ol tag
ج) br tag
د) p tag

54 اختر ناتج تنفيذ البرنامج التالي:

```
<ul>
  <li>strawberry</li>
  <li>cherry</li>
  <li>orange</li>
</ul>
```



أ strawberry cherry orange
ب 1.strawberry 2.cherry 3.orange
ج .strawberry .cherry .orange
د (strawberry) (cherry) (orange)

55 اختر ناتج تنفيذ البرنامج التالي:

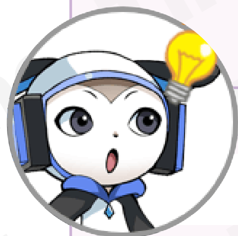
```
<ol>
  <li>Cut ingredients</li>
  <li>Stir-fry</li>
  <li>Add seasonings</li>
</ol>
```



أ 1.Cut ingredients 2.Stir-fry 3.Add seasonings
ب .Cut ingredients .Stir-fry .Add seasonings
ج Cut ingredients Stir-fry Add seasonings
د (Cut ingredients) (Stir-fry) (Add seasonings)

56 ما الطريقة الصحيحة لإضافة تعليق (Comment) في لغة HTML؟

أ <!-- comment -->
ب - comment -
ج // comment //
د <comment>



57 اختر الوسم لعرض صورة باسم الصورة image.png.

أ <img="image.png">
ب <img><"image.png">
ج <src="image.png">
د <img src="image.png">

58 اختر الطريقة الصحيحة لكتابة الخاصية alt التي تسمح لك بإضافة وصف للصورة.

أ <img><"This is an image.">
ب <img="This is an image.">
ج <img alt="This is an image.">
د <alt="This is an image.">



59 لغة أنماط تستخدم لتعديل جوانب مختلفة من "المظهر والشكل" لموقع الويب؟

أ HTML
ب CSS
ج C++
د JavaScript



60 اختر الوصف الصحيح لخاصية color.

أ تغيير لون النص
ب تغيير لون الخلفية
ج تغيير لون الصورة
د تغيير حجم الخط

61 اختر الوصف الصحيح لخاصية background-color.

أ تغيير لون النص
ب تغيير لون الخلفية
ج تغيير سُمك الخط
د تغيير لون الصورة

62 اختر الوصف الصحيح لقيمة خاصية font-weight عند كتابتها كـ bold.

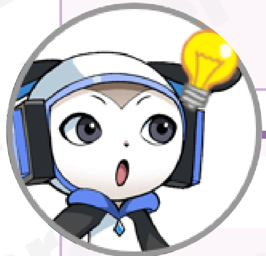
أ يصبح حجم الخط أصغر
ب يتغير لون النص إلى ذهبي
ج يصبح النص نحيفًا
د يصبح النص عريضًا

63 اختر القيمة الصحيحة لخاصية font-size.

أ font-size: 10;
ب font-size: 10size;
ج font-size: 10g;
د font-size: 10px;

64 ماذا تُسمى علامة "&"؟

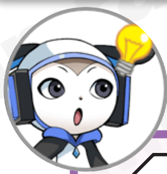
أ Condition
ب Array
ج Ampersand
د Vertical bar



65 ماذا تُسمى علامة "|"؟

أ Condition
ب Array
ج Ampersand
د Vertical bar





2 أكمل ما يأتي بما هو مناسب:....

1 اعرض "Good morning" في وحدة التحكم.

JavaScript

....

2 اعرض "Programming" في وحدة التحكم.

JavaScript

```
console.log(                    );
```

3 علّق البرنامج بحيث تكون المخرجات كما يلي:

Hello.
My name is
Cody.

JavaScript

```
console.log("Hello.");
console.log("My name is");
console.log("Rick.");
console.log("Cody.");
```

4 اجعل الكمبيوتر يحسب 19+40 ويعرض نتيجةالحساب في وحدة التحكم.

JavaScript

....

5 اجعل الكمبيوتر يحسب 7-2 ويعرض نتيجة الحساب في وحدة التحكم.

JavaScript

....

6 اجعل الكمبيوتر يحسب 15 ÷ 5 ويعرض نتيجة الحساب في وحدة التحكم.

JavaScript

....

7 اجعل الكمبيوتر يحسب 5*2 ويعرض "The answer is 10." في وحدة التحكم.

JavaScript

....

8 أنشئ برنامج للإعلان عن المتغير number .

JavaScript

....



9) أسند 100 إلى المتغير number .

JavaScript

```
..........................................................................
```

10) 1) أسند قيمة المتغير number مضافاً إليها 10 .
2) اعرض قيمة المتغير number في وحدة التحكم.

JavaScript

```
..........................................................................
..........................................................................
```

11) 1) أعلن عن المتغير num وأسند له 100.
2) إذا كان المتغير num يساوي 100، اطبع "Correct".

JavaScript

```
let ............ = .......... ;
if( .................... ){console.log ( .................... );}
```

12) 1) صرح عن المتغير num وأسنِد له القيمة 5.
2) إذا كان المتغير num يساوي 10، أخرج "Win".
3) وإلا، أخرج "Lose".

JavaScript

```
let ............ = .......... ;
..........................................................................
else { .................................... }
```

13) 1) صرح عن المتغير score وأسنِد له القيمة 80 .
2) إذا كان المتغير score أكبر من أو يساوي 60، أخرج "Pass".
3) وإلا، أخرج "Fail".

JavaScript

```
..........................................................................
..........................................................................
..........................................................................
```

14) أكمل في السطر الثالث:
إذا لم يكن كذلك، وإذا كانت قيمة المتغير num هي 2،
اعرض "Two".

JavaScript

```
let num = 2;
if (num == 1){console.log("One");}
else if ..................................................................
```

15 أكمل في السطر الثالث:
إذا لم يكن كذلك، وإذا كانت قيمة المتغير score أكبر من أو تساوي 80، اعرض "Pass".

JavaScript
```
let score = 82;
if (score == 100){console.log("Perfect score");}
else if ........................................
```

16 (1) إذا كان المتغير num أصغر من أو يساوي 3، اطبع "A".
(2) إذا لم يكن كذلك، وإذا كان num أكبر من أو يساوي 6، اطبع "B".
(3) إذا لم يكن كذلك، اطبع "C".

JavaScript
```
let num = 4;
if (                ){console.log("A");}
else if (                ){console.log("B");}
        {console.log("C");}
```

17 1. أعلن عن المتغير egg وأسند له 5.
2. إذا كان المتغير egg يساوي 0، اعرض "No stock".
3. إذا لم يكن كذلك، وإذا كان المتغير egg أقل من أو يساوي 3، اعرض "Low stock".
4. إذا لم يكن كذلك، اعرض "In stock".

JavaScript
```
........................................
........................................
........................................
........................................
```

18 الرجاء إنشاء برنامج كما يلي.
1. إذا كان المتغير a يساوي 5 أو b يساوي 10، اطبع "Condition met".
*أدخل التعبير الشرطي فقط.

JavaScript
```
let a = 5;
let b = 10;
if (                ){console.log("Condition met");}
```

19 الرجاء إنشاء برنامج كما يلي.
1. إذا كان المتغير a أكبر من أو يساوي 2 و b أكبر من أو يساوي 3، اطبع "Condition met". *أدخل التعبير الشرطي فقط.

JavaScript
```
let a = 4;
let b = 3;
if (                ){console.log("Condition met");}
```



20 أخرج "Hello" إلى وحدة التحكم 7 مرات.
*اسم المتغير i وأعطه القيمة 0 في البداية.
*استخدم عامل الزيادة (increment operator).

JavaScript
```
     (                    ) {
  console.log("Hello");
 }
```

21 الرجاء إضافة برنامج يستدعي الدالة calc.
ضع 3 كقيمة للمعامل.

JavaScript
```
function calc(num) {
  let result = num * 5;
  return result;
}
console.log(                    );
```

22 الرجاء إضافة برنامج يعيد قيمة المتغير result إلى البرنامج الموصوف.

JavaScript
```
function calc(num) {
  let result = num * 7;

}
console.log(calc(10));
```

23 (1) استخدم الدالة للتقريب لـ 5.5 وأخرج الناتج إلى وحدة التحكم.
(2) استخدم الدالة للتقريب إلى أسفل للعدد 8.8 إلى أقرب عدد صحيح وأخرج الناتج إلى وحدة التحكم.

JavaScript
```



```

24 أنشئ برنامج يقوم بإخراج "حظ ممتاز" إذا كانت القيمة العشوائية أكبر من أو تساوي 0.5، وإخراج "حظ سيئ" في غير ذلك.
استخدم الدالة Math.random() وأدخل عبارة الشرط if كانت القيمة العشوائية أكبر من أو تساوي 0.5.

JavaScript
```
if (                    ) {
  console.log("Excellent luck");}
else {console.log("Bad luck");}
```



25 أنشئ برنامجاً يعيد عدداً صحيحاً عشوائياً كما يلي:
(1) استخدم الدالة Math.random لإرجاع قيمة عددية عشوائية.
(2) اضرب القيمة العشوائية الناتجة في 10.
(3) استخدم الدالة Math.floor لإرجاع عدد صحيح.
يجب أن يكون البرنامج في سطر واحد.

JavaScript

....

26 أنشئ برنامجك كما يلي.
- حدد نوع البيانات لـ `"7"` في الحاسوب واطبعه في وحدة التحكم.

JavaScript

....

27 قم بالاعلان عن مصفوفة باسم num وضع القيم التالية بداخلها
5، 10، 15

JavaScript

....

28 اطبع جميع القيم في المصفوفة على وحدة التحكم.

JavaScript

```
let num = [5, 10, 15, 20, 25];
```

....

29 اطبع طول المصفوفة على وحدة التحكم.

JavaScript

```
let num = [5, 10, 15, 20, 25];
```

....

30 الرجاء كتابة تعليق على السطر الأول من النص.

HTML

```
      The h1 tag is a heading.
<h1>Hello</h1>
```

31 الرجاء كتابة Hello كعنوان رئيسي و Welcome كأصغر عنوان .

HTML

....

....



32 حدد الـ selector كـ "h1".

CSS
```
        {
 color: red;}
```

33 حدد القيمة "blue".

CSS
```
h1 {
  color:            ;}
```

34 يُحدد اسم الـ class كما يلي:

<p class="dog">Dog</p>
<p class="cat">Cat</p>

اكتب CSS لتغيير كلمة "Dog" إلى اللون الأزرق.

CSS
```


```

35 اترك تعليقًا في السطر الأول من النص.

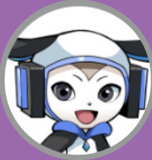
CSS
```
   Change the text color to navy.
   p {
     color: navy;}
```

نهاية أسئلة المنصة





الإجابات الصحيحة

1 اختر الإجابة الصحيحة من بين الإختيارات التالية :

رقم السؤال	الإجابة الصحيحة	رقم السؤال	الإجابة الصحيحة	رقم السؤال	الإجابة الصحيحة	رقم السؤال	الإجابة الصحيحة	رقم السؤال	الإجابة الصحيحة
1	ب	14	أ	27	د	40	ب	53	ب
2	ب	15	د	28	ج	41	د	54	ج
3	أ	16	ج	29	أ	42	أ	55	أ
4	ج	17	ب	30	ج	43	ب	56	أ
5	د	18	أ	31	أ	44	ج	57	د
6	ب	19	ج	32	ب	45	د	58	ج
7	ب	20	ب	33	ج	46	د	59	ب
8	ب	21	ب	34	د	47	أ	60	أ
9	أ	22	د	35	د	48	د	61	ب
10	ج	23	د	36	ج	49	أ	62	د
11	د	24	أ	37	أ	50	ب	63	د
12	ج	25	أ	38	أ	51	ج	64	ج
13	أ	26	ج	39	أ	52	أ	65	د



2 أكمل ما يأتي بما هو مناسب:....

<table>
<tr><th>رقم السؤال</th><th>الإجابة الصحيحة</th></tr>
<tr><td>1</td><td>console.log(“Good morning“);</td></tr>
<tr><td>2</td><td>“Programming“</td></tr>
<tr><td>3</td><td>//console.log(“Rick“);</td></tr>
<tr><td>4</td><td>console.log(19+40);</td></tr>
<tr><td>5</td><td>console.log(7-2);</td></tr>
<tr><td>6</td><td>console.log(15/5);</td></tr>
<tr><td>7</td><td>console.log(“The answer is “ + (2 * 5) + “.”);</td></tr>
<tr><td>8</td><td>let number;</td></tr>
<tr><td>9</td><td>number = 100;</td></tr>
<tr><td>10</td><td>number = number + 10;
console.log(number);</td></tr>
<tr><td>11</td><td>let num = 100;
if (num == 100){console.log(“Correct”)};</td></tr>
<tr><td>12</td><td>let num = 5;
if (num == 10){console.log(“Win”);}
else{console.log(“Lose”);}</td></tr>
<tr><td>13</td><td>let score = 80;
if (score >= 60){console.log(“Pass”);}
else {console.log(“Fail”);}</td></tr>
<tr><td>14</td><td>(num == 2){console.log(“Two”);}</td></tr>
<tr><td>15</td><td>(score >= 80){console.log(“Pass”);}</td></tr>
<tr><td>16</td><td>(num <= 3)
(num >= 6)
else</td></tr>
<tr><td>17</td><td>let egg = 5;
if (egg == 0){console.log(“Nostock”);}
else if (egg <= 3){ console.log(“Low stock”);}
else {console.log(“In stock”);}</td></tr>
<tr><td>18</td><td>(a == 5 || b == 10)</td></tr>
<tr><td>19</td><td>(a >= 2 && b >= 3)</td></tr>
<tr><td>20</td><td>for (let i = 0; i < 7; i ++)</td></tr>
<tr><td>21</td><td>(calc(3));</td></tr>
<tr><td>22</td><td>return result;</td></tr>
<tr><td>23</td><td>console.log(Math.round(5.5));
console.log(Math.floor(8.8));</td></tr>
<tr><td>24</td><td>(Math.random() >= 0.5)</td></tr>
<tr><td>25</td><td>console.log(Math.floor(Math.random() * 10));</td></tr>
<tr><td>26</td><td>console.log(typeof “7”);</td></tr>
<tr><td>27</td><td>let num = [5, 10, 15];</td></tr>
<tr><td>28</td><td>console.log(num);</td></tr>
<tr><td>29</td><td>console.log(num.length);</td></tr>
<tr><td>30</td><td><!--The h1 tag is a heading.--></td></tr>
<tr><td>31</td><td><h1>Hello</h1>
<h6>Welcome</h6></td></tr>
<tr><td>32</td><td>h1{
  color: red;
}</td></tr>
<tr><td>33</td><td>h1 {
  color:blue;
}</td></tr>
<tr><td>34</td><td>.dog{
  color: blue;
}</td></tr>
<tr><td>35</td><td>/*Change the text color to navy.*/</td></tr>
</table>



كيفية طباعة صفحات معينة من ملف معين

مثلا ازاى نطبع الصفحات من صفحة 4 الى صفحة 9

